

1996-265715 JAPIO  
 TI DATA BROADCASTING RECEIVER  
 IN HAYASHI TERUTAKA; TOKUTANI SHIGENOBU; OCHI KEIICHI  
 PA NINTENDO CO LTD, JP (CO 352321)  
 PI JP08265715 A 19961011 Heisei  
 AI JP1995-65636 (JP07065636 Heisei) 19950324  
 SO PATENT ABSTRACTS OF JAPAN (CD-ROM), Unexamined Applications, Vol. 96,  
 No. 10  
 IC ICM (6) H04N007-08  
 ICS (6) H04N007-081; (6) G06F011-00; (6) G06F013-00; (6) G06F013-00;  
 (6) H04B001-16; (6) H04H001-00; (6) H04L001-00  
 CC 44.6 COMMUNICATION - Television  
 30.2 MISCELLANEOUS GOODS - For athletic and amusement  
 34.4 SPACE DEVELOPMENT - Space communication  
 44.3 COMMUNICATION - Telegraphy  
 44.5 COMMUNICATION - Radio broadcasting  
 45.1 INFORMATION PROCESSING - Arithmetic sequence units  
 45.2 INFORMATION PROCESSING - Memory unit  
 CT R005 COMMON - Piezoelectric ferroelectric  
 R116 ELECTRONIC MATERIAL - Light emitting diode (LED)  
 AB PURPOSE: To provide a data broadcasting receiver capable of checking  
 errors without burdening a CPU so much.  
 CONSTITUTION: Bit stream signals inputted from a BS tuner are  
 converted to  
 the data signals of a packet unit and then, separated to prefix data  
 and  
 data group data. Here, the presence/absence of the errors of the data  
 group data are detected by an error detection circuit, an error flag  
 is  
 inserted to the prefix data and they are stored in a prefix memory  
 and a  
 data memory. In the case where the respective data stored in the  
 prefix  
 memory and the data memory are transferred to a memory circuit to be  
 used  
 in a video game machine, an OR loop 483 ORs the error flag read out  
 of the  
 prefix memory in the time base direction and stores the result in a  
 D-FF  
 486. The CPU refers to the stored ORed result and confirms whether or  
 not  
 the error is present in the data group data.



## PATENT ABSTRACTS OF JAPAN

(11) Publication number: **08265715 A**

(43) Date of publication of application: **11.10.96**

(51) Int. Cl

H04N 7/08  
H04N 7/081  
G06F 11/00  
G06F 13/00  
G06F 13/00  
H04B 1/16  
H04H 1/00  
H04L 1/00

(21) Application number: 07065636

(22) Date of filing: 24.03.95

(71) Applicant: **NINTENDO CO LTD**

(72) Inventor: HAYASHI TERUTAKA  
TOKUTANI SHIGENOBU  
OCHI KEIICHI

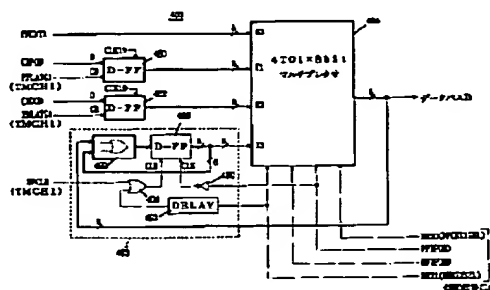
**(54) DATA BROADCASTING RECEIVER**

(57) Abstract:

**PURPOSE:** To provide a data broadcasting receiver capable of checking errors without burdening a CPU so much.

**CONSTITUTION:** Bit stream signals inputted from a BS tuner are converted to the data signals of a packet unit and then, separated to prefix data and data group data. Here, the presence/absence of the errors of the data group data are detected by an error detection circuit, an error flag is inserted to the prefix data and they are stored in a prefix memory and a data memory. In the case where the respective data stored in the prefix memory and the data memory are transferred to a memory circuit to be used in a video game machine, an OR loop 483 ORs the error flag read out of the prefix memory in the time base direction and stores the result in a D-FF 486. The CPU refers to the stored ORed result and confirms whether or not the error is present in the data group data.

COPYRIGHT: (C)1996,JPO





## 【特許請求の範囲】

【請求項 1】 受信されたビットストリームのデータ信号を入力して、情報処理装置で処理可能なデータに変換するデータ放送受信装置であって、前記ビットストリーム信号をパケット単位のデータ信号に変換するデータ変換手段、前記パケット単位のデータ信号から、パケットの属性を示す第 1 のデータと、番組の内容に対応する第 2 のデータとを分離する分離手段、前記分離手段によって分離された第 2 のデータにエラーがあるか否かを検出するエラー検出手段、前記分離手段によって分離された第 1 および第 2 のデータを、一時的に記憶するバッファリング手段、前記エラー検出手段の検出結果を示すエラーフラグを前記第 1 のデータに挿入することにより、前記バッファリング手段に記憶させるエラーフラグ挿入手段、前記バッファリング手段に記憶されている前記第 1 および第 2 のデータを、前記情報処理装置で使用されるメモリ回路に転送する際に、当該バッファリング手段から読み出されたエラーフラグを時間軸方向に論理和演算する論理和手段、および前記論理和手段の演算結果に基づいて、前記バッファリング手段から読み出されて転送される第 2 のデータに 1 つでもエラーが生じているか否かを確認するエラー確認手段を備える、データ放送受信装置。

【請求項 2】 前記バッファリング手段に記憶された前記エラーフラグは、前記第 1 および第 2 のデータと共に、前記情報処理装置で使用されるメモリ回路に転送して格納されることを特徴とする、請求項 1 に記載のデータ放送受信装置。

【請求項 3】 前記第 1 のデータは、データの区切りを示す区切り用識別子を含み、前記論理和手段は、前記エラーフラグと共に、前記第 1 のデータを時間軸方向に論理和演算し、前記論理和手段の演算結果に基づいて、前記バッファリング手段に記憶された前記第 2 のデータに、先頭パケットまたは終端パケットが存在するか否かを確認するパケット確認手段をさらに備える、請求項 1 に記載のデータ放送受信装置。

【請求項 4】 放送衛星から受信したビットストリームのデータ信号を入力して、情報処理装置で処理可能なデータに変換することを特徴とする、請求項 1 ～ 3 のいずれかに記載のデータ放送受信装置。

## 【発明の詳細な説明】

## 【0001】

【産業上の利用分野】 本発明は、データ放送受信装置に関し、より特定的には、受信されたビットストリームのデータ信号を入力して、情報処理装置で処理可能なデータに変換するためのデータ放送受信装置に関する。

## 【0002】

【従来の技術】 衛星通信技術の急速な発達に伴い、最近、衛星によるテレビジョン放送のデータチャンネルを利用して、各種のデジタル信号を伝送するような、衛星データ放送システムが注目されつつある。そのため、かかるシステムの標準化のための様々な討議がなされている（例えば、電気通信審議会衛星データ放送委員会報告概要の資料 67-13、または電気通信審議会衛星データ放送委員会報告の資料 67-14 参照）。

【0003】 ところで、衛星によるデータ放送は、天気によって左右されやすく、降雨や雷等によって様々なエラーが伝送パケット中に発生する。そのため、従来の衛星放送受信システムでは、受信した各データパケット毎にエラー発生の有無を検出し、パケット単位でエラー処理を行っていた。

## 【0004】

【発明が解決しようとする課題】 上記のように、従来は、エラー発生の有無にかかわらず、常に全パケットのエラーの有無を検出するようにしているので、CPU に余計な負担がかかるという問題点があった。

【0005】 それゆえに、本発明の目的は、CPU にそれほど負担をかけることなくエラーチェックが行えるデータ放送受信装置を提供することである。

## 【0006】

## 【課題を解決するための手段】

【0007】 請求項 1 に係る発明は、受信されたビットストリームのデータ信号を入力して、情報処理装置で処理可能なデータに変換するデータ放送受信装置であって、ビットストリーム信号をパケット単位のデータ信号に変換するデータ変換手段、パケット単位のデータ信号から、パケットの属性を示す第 1 のデータと、番組の内容に対応する第 2 のデータとを分離する分離手段、分離手段によって分離された第 2 のデータにエラーがあるか否かを検出するエラー検出手段、分離手段によって分離された第 1 および第 2 のデータを、一時的に記憶するバッファリング手段、エラー検出手段の検出結果を示すエラーフラグを第 1 のデータに挿入することにより、バッファリング手段に記憶させるエラーフラグ挿入手段、バッファリング手段に記憶されている第 1 および第 2 のデータを、情報処理装置で使用されるメモリ回路に転送する際に、当該バッファリング手段から読み出されたエラーフラグを時間軸方向に論理和演算する論理和手段、および論理和手段の演算結果に基づいて、バッファリング手段から読み出されて転送される第 2 のデータに 1 つでもエラーが生じているか否かを確認するエラー確認手段を備えている。

【0008】 請求項 2 に係る発明は、請求項 1 の発明において、バッファリング手段に記憶されたエラーフラグは、第 1 および第 2 のデータと共に、情報処理装置で使用されるメモリ回路に転送して格納されることを特徴とする。

【0009】請求項3に係る発明は、請求項1の発明において、第1のデータは、データの区切りを示す区切り用識別子を含み、論理和手段は、エラーフラグと共に、第1のデータを時間軸方向に論理和演算し、論理和手段の演算結果に基づいて、バッファリング手段に記憶された第2のデータに、先頭パケットまたは終端パケットが存在するか否かを確認するパケット確認手段をさらに備えている。

【0010】請求項4に係る発明は、請求項1～3のいずれかの発明において、放送衛星から受信したビットストリームのデータ信号を入力して、情報処理装置で処理可能なデータに変換することを特徴とする。

【0011】

【作用】請求項1に係る発明では、パケット単位のデータ信号から分離された第1のデータ（パケットの属性を示すデータ）および第2のデータ（番組の内容に対応するデータ）をバッファリング手段に格納する際に、第2のデータにエラーがあるか否かをエラー検出手段によって検出し、このエラー検出手段の検出結果を示すエラーフラグを第1のデータに挿入してバッファリング手段に格納するようにしている。そして、バッファリング手段に記憶されている第1および第2のデータを、情報処理装置で使用されるメモリ回路に転送する際に、当該バッファリング手段から読み出されたエラーフラグを時間軸方向に論理和演算して、後に参照することにより、バッファリング手段からメモリ回路に転送される第2のデータ中に1つでもエラーが生じているか否かを確認するようにしている。このように、パケット単位でエラーをチェックすることなく、転送データにエラーが生じているか否かを容易に確認することができる。従って、エラーチェックに要するCPUの負担を軽減することができる。

【0012】請求項2に係る発明では、バッファリング手段に記憶されたエラーフラグは、第1および第2のデータと共に、情報処理装置で使用されるメモリ回路に転送して格納される。これによって、CPUは、エラー確認手段がエラーの存在を確認したとき、メモリ回路に格納されたエラーフラグを参照することで、どのパケットにエラーが生じているかを容易に知ることができる。

【0013】請求項3に係る発明では、論理和手段は、エラーフラグと共に、第1のデータに含まれている区切り用識別子を時間軸方向に論理和演算する。そして、パケット確認手段は、論理和手段の演算結果に基づいて、バッファリング手段に記憶された第2のデータに、先頭パケットまたは終端パケットが存在するか否かを確認する。これによって、CPUは、パケット単位で区切り用識別子をチェックすることなく、データの先頭と終端とを知ることができる。

【0014】請求項4に係る発明では、放送衛星から受信したビットストリームのデータ信号を入力して、情報

処理装置で処理可能なデータに変換するようにしている。前述したように、衛星によるデータ放送は、天候等により様々なエラーが発生し易いので、本発明のエラーチェックによるCPUの負担軽減効果が特に顕著である。

【0015】

【実施例】図1は、本発明の一実施例に係る衛星データ放送受信装置の構成を示す外観斜視図である。図1において、この衛星データ放送受信装置は、ゲーム機本体2と、衛星データ放送受信アダプタ4と、衛星データ放送受信カートリッジ6と、メモリユニット8と、ハードディスク装置10とを備えている。ハードディスク装置10は、衛星データ放送受信アダプタ4に装着される。ゲーム機本体2は、衛星データ放送受信アダプタ4の上に搭載される。メモリユニット8は、衛星データ放送受信カートリッジ6に装着される。衛星データ放送受信カートリッジ6は、ゲーム機本体2に装着される。ゲーム機本体2は、いわゆるテレビゲーム機であって、例えば、本出願人の製造販売に係る「スーパーファミコン（商品名；登録商標）」が用いられる。

【0016】図2は、本実施例の衛星データ放送受信装置を用いて構成された衛星データ放送受信システムの構成を示す外観斜視図である。図2において、ゲーム機本体2には、コントローラ12と、TVモニタ22とが接続される。衛星データ放送受信アダプタ4には、ACアダプタ14が接続されるとともに、ビットストリーム分配器20を介して、BSチューナ18が接続される。BSチューナ18には、BSアンテナ16が接続される。ビットストリーム分配器20は、BSチューナ18からの放送データ（ビットストリーム）を、衛星データ放送受信アダプタ4に与えるとともに、他の衛星データ受信機器に対して当該信号を分配して供給する。

【0017】図3は、図2におけるゲーム機本体2の内部構成を示すブロック図である。図3において、このゲーム機本体2は、CPU30と、ワークRAM32と、アドレスデコーダ34と、入出力ポート36と、コネクタA38と、ビデオプロセッサ（PPU）40と、ビデオRAM42と、ビデオエンコーダ44と、オーディオプロセッサ（APU）46と、オーディオRAM48と、ステレオサウンドの左用ミキサ50aと、右用ミキサ50bと、左用アンプ52aと、右用アンプ52bと、コネクタB54と、CIC56と、リセットスイッチ58と、リセット回路60と、パワースイッチ62と、電源回路64と、発振回路66とを備えている。

【0018】CPU30は、DMA回路30aを内蔵しており、DMA動作時にはアドレスバスAおよびBに異なるアドレス信号を与え、アドレスバスAに接続された回路からアドレスバスBに接続された回路へ（または逆方向に）データを転送する。CPUは、コネクタA38からの割り込み信号IRQBおよびコネクタB54から

の割り込み信号 I R Q B がワイヤード OR 接続されて入力される。

【0019】ワーク RAM 3 2 は、基本的にはアドレスバス A 上のメモリとして使用されるが、主としてプログラム動作上の補助情報を記憶するものであり、アドレスバス B 上のポートからもアクセスが可能である。アドレスデコーダ 3 4 は、アドレスバス A 上の信号に基づいて、各回路に対してチップイネーブル信号を供給する。入出力ポート 3 6 は、コントローラ 1 2 から操作情報を得る。コネクタ A 3 8 は、衛星データ放送受信カートリッジ 6 に接続される。このコネクタ A 3 8 には、アドレスバス A および B の両方が接続される。

【0020】PPU 4 0 は、画像データの処理を行うものであり、作業用メモリとして用いられるビデオ RAM 4 2 が接続される。ビデオエンコーダ 4 4 は、PPU 4 0 からのアナログ RGB 信号をコンポジットビデオ信号に変換し、TV モニタ 2 2 に供給する。

【0021】APU 4 6 は、音声データの処理を行う。オーディオ RAM 4 8 は、APU 4 6 の作業用メモリとして用いられる。ミキサ 5 0 a および 5 0 b は、APU 4 6 からの音声信号と、衛星からの PCM 音声放送とを混合して出力する。アンプ 5 2 a および 5 2 b は、それぞれ、ミキサ 5 0 a および 5 0 b の出力信号を増幅し、TV モニタ 2 2 に供給する。

【0022】コネクタ B 5 4 は、衛星データ放送受信アダプタ 4 に接続される。このコネクタ B 5 4 には、アドレスバス B は接続されるが、アドレスバス A は接続されない。C I C 5 6 は、カートリッジの真性を検査するためのセキュリティチップであり、その詳細が特公平 5 - 2 7 8 9 1 号公報に詳細に開示されている。C I C 5 6 は、カートリッジが真正でない場合に、リセット信号を出力する。リセットスイッチ 5 8 は、操作者によって操作されたときに、リセット信号を出力する。リセット回路 6 0 は、C I C 5 6 からのリセット信号とリセットスイッチ 5 8 からのリセット信号とを混合し、CPU 3 0、PPU 4 0、APU 4 6、コネクタ A 3 8 およびコネクタ B 5 4 に、リセット信号 R E S E T B として供給する。また、リセット回路 6 0 は、パワースイッチ 6 2 が ON されたときにも、リセット信号 R E S E T B を出力する。

【0023】パワースイッチ 6 2 は、操作者によって操作されたときに、電源回路 6 4 を能動化する。電源回路 6 4 は、衛星データ放送受信アダプタ 4 からの供給電圧 V D C ( 1 2 V ) を 5 V に変換し、安定化させる。この 5 V の電圧は、電源ライン M V c c を介して各回路およびコネクタに供給される。特にコネクタ B 5 4 を介して衛星データ放送受信アダプタ 4 に供給される 5 V 電圧は、電源制御信号として使用される。発振回路 6 6 は、各プロセッサに、必要なクロック信号を与える。

【0024】図 4 は、図 2 における衛星データ放送受信

カートリッジ 6 の内部構成を示すブロック図である。図 4 において、この衛星データ放送受信カートリッジ 6 は、マッピングコントローラ 7 0 と、マスク ROM 7 2 と、疑似 SRAM 7 4 と、フラッシュメモリ A 7 6 と、C I C 7 8 と、コネクタ D 8 0 とを備えている。

【0025】マッピングコントローラ 7 0 は、プログラムに従って、メモリマップ上における、マスク ROM 7 2、疑似 SRAM 7 4、フラッシュメモリ A 7 6 およびメモリユニット 8 (コネクタ D 8 0 に接続される) の配置を切り替える。マッピングコントローラ 7 0 の内部の詳細は、図 7 に示されている。

【0026】マスク ROM 7 2 は、主として放送受信プログラムや、初期画面表示等の基本的プログラムおよびデータを記憶する。CPU 3 0 のリードタイミングを示す信号 C P U R D B およびマッピングコントローラ 7 0 からのチップイネーブル信号 C E 2 B が、それぞれ、アウトプットイネーブルコントロール端子 O E 1 および O E 2 に与えられる。

【0027】疑似 SRAM 7 4 は、主として放送データを一時記憶するものであり、マッピングコントローラ 7 0 からの信号 P W E B によってリード/ライト制御が行われ、信号 P O E B / R F B によってアウトプットイネーブルおよびリフレッシュ制御が行われ、信号 C E 1 B によってチップイネーブル制御が行われる。

【0028】フラッシュメモリ A 7 6 は、主として放送データを長期的に記憶するものであり、CPU 3 0 からの信号 C P U R D B によってアウトプットイネーブル制御が行われ、信号 C P U W R B でリード/ライト制御が行われる。また、フラッシュメモリ A 7 6 は、マッピングコントローラ 7 0 からの信号 W P B でライトプロテクト制御が行われ、信号 C E 3 B でチップイネーブル制御が行われる。CPU 3 0 からのデータ書き込み時には、信号 R Y / B Y B がローレベルからハイレベルに変化することにより、書き込み完了をマッピングコントローラ 7 0 に知らせる。これに基づいてマッピングコントローラ 7 0 は、CPU 3 0 に対して信号 I R Q B を発生する。C I C 7 8 は、ゲーム機本体 2 内の C I C 5 6 と協働して、セキュリティ動作を行う。コネクタ D 8 0 は、メモリユニット 8 に接続される。

【0029】図 5 は、図 2 における衛星データ放送受信アダプタ 4 の内部構成を示すブロック図である。図 5 において、この衛星データ放送受信アダプタ 4 は、データ放送デコーダ 8 2 と、PCM デコーダ 8 4 と、左用アンプ 8 6 a と、右用アンプ 8 6 b と、EEPROM 8 8 と、ボルテージ・コントロールド・クリスタル・オシレータ (以下、「V X O」と略称する) 9 0 と、双方向バッファ 9 2 と、バスバッファ 9 4、9 6 と、コネクタ C 9 8 と、LED 1 0 0 と、電源制御回路 1 0 2 と、LED 1 0 4 とを備えている。

【0030】データ放送デコーダ 8 2 は、ビットストリ

ーム分配器 20 を介して、BS チューナ 18 からビットストリーム信号 BST を入力し、この信号を増幅した後には信号 BST02 として PCM デコーダ 84 に与える。PCM デコーダ 84 から出力される信号 FDI N は、ビットストリーム信号から分離された放送データであり、データ放送デコーダ 82 は、この信号 FDI N をデコードしてゲーム機本体 2 の CPU 30 が使用可能な形態のデータに変換する。信号 VER は、当該アダプタ 4 のバージョン番号を設定するために用いられ、本実施例では 0 に設定されている。信号 IRQB は、例えばハードディスク装置 10 からの割り込み要求が発生した場合に発生され、コネクタ B54 を介して CPU 30 に与えられる。データ放送デコーダ 82 の内部の詳細は、図 12 に示されている。

【0031】PCM デコーダ 84 (例えば、松下電器産業株式会社製の MN88821) は、データ放送デコーダ 82 から与えられる信号 BST02 から PCM 音声信号 PCML, PCMR を取り出し、アンプ 86a、86b を介して、ゲーム機本体 2 内のミキサ 50a、50b に供給する。また、PCM デコーダ 84 は、フレーム同期信号 FSYNC、ビットストリームクロック信号 BSCK を生成してデータ放送デコーダ 82 に供給するとともに、放送データをシリアル信号 FDI N としてデータ放送デコーダ 82 に供給する。プログラムによる PCM デコーダ 84 の制御は、PCMD バスの信号を介して行われる。データ放送デコーダ 82 は、PCMD バスを CPU 30 に結合するための入出力ポートを内蔵する。

【0032】EEP-ROM 88 (例えば、セイコー電子株式会社製の S-2913CR) には、本体のシリアル番号が出荷時に記録されている。有料データ受信時は、この本体シリアル番号が参照され、有料放送受信登録ユーザーであるかどうか判断される。EEP-ROM 88 に対する書き込み、読み出しの制御は、EEP-ROM バスの信号を介してソフトウェアにより行われる。データ放送デコーダ 82 は、EEP-ROM バスを CPU 30 に結合するための入出力ポートを内蔵する。

【0033】VXO90 (例えば、松下電器産業株式会社製の AN3913S) は、受信されたビットストリーム信号に同期したメインクロック信号 MCK を発生する。メインクロック信号 MCK は PCM デコーダ 84 に与えられ、これに基づいてビットストリームクロック信号 BSCK が生成される。データ放送デコーダ 82 は、ビットストリームクロック信号 BSCK とビットストリーム信号との位相を比較し、それらの位相誤差成分を信号 P/D として VXO90 にフィードバックをかける。これによってメインクロック信号 MCK、ビットストリームクロック信号 BSCK およびビットストリーム信号の同期が達成される。信号 STDBY は、プログラム制御のスタンバイ信号であり、スタンバイ時には VXO90 の発振が停止して消費電力が低減される。

【0034】双方向バッファ 92 (例えば、米国テキサスインスツルメント社製の 74LS245) は、データバス用のバスバッファである。この双方向バッファ 92 は、データ放送デコーダ 82 からの信号 DIR によって、信号の向きが切り替えられる。すなわち、双方向バッファ 92 は、CPU 30 がデータを入力する場合にはコネクタ B54 方向にデータを出力し、CPU 30 がデータを出力する場合にはデータ放送デコーダ 82 およびコネクタ C98 方向にデータを出力するよう制御される。

【0035】バスバッファ 94 (例えば、米国テキサスインスツルメント社製の 74LS541) はアドレスバス用のバスバッファであり、バスバッファ 96 (例えば、米国テキサスインスツルメント社製の 74LS541) は制御信号用のバスバッファである。コネクタ C98 には、例えばハードディスク装置 10 が接続され、数多くの受信データを長期間に渡って保存するのに用いられる。ハードディスク装置 10 と各メモリ間のデータ転送は、データバス、アドレスバス B を介して制御信号 B に基づいて DMA 回路 30a により制御されるが、ハードディスク装置 10 への命令は、EX バスを介してソフトウェアにより与えられる。データ放送デコーダ 82 は、EX バスを CPU 30 に結合するための入出力ポートを内蔵する。

【0036】LED (発光ダイオード) 100 は、ハードディスク装置 10 のアクセスランプ等に使用される。電源制御回路 102 は、衛星データ放送受信アダプタ 4 の電源を制御する。ゲーム機本体 2 から電源ライン MVcc を介して 5V 電圧が供給されると、電源制御回路 102 は、AC アダプタ 14 から供給される 12V 電源を 5V 電圧に変換し、安定化して SVcc として、アダプタ 4 内の各回路およびコネクタに電源を供給する。LED 104 は、パワーオン時に点灯するパイロットランプである。

【0037】図 6 は、図 2 におけるメモリユニット 8 の内部構成を示すブロック図である。図 6 において、このメモリユニット 8 は、フラッシュメモリ B106 を備えている。このフラッシュメモリ B106 は、主として長期的記憶に用いられ、他のカートリッジに差し替えて使用することにより、受信データを他のカートリッジでも利用可能になる。また、データ放送を受信できるカートリッジが複数個ある場合でもメモリユニット 8 を差し替えて使用できるため、カートリッジの単価を安くすることができる。なお、フラッシュメモリ B106 に接続された各信号線は、図 4 におけるフラッシュメモリ A76 に接続されたものと同等である。

【0038】図 7 は、図 4 におけるマッピングコントローラ 70 の内部構成を示すブロック図である。図 7 において、このマッピングコントローラ 70 は、コントロールレジスタ 110 と、アドレスデコーダ 112 と、NA

NDゲート114と、スリーステートゲート116と、入力バッファ118と、マップデコーダ120と、アドレスセクタ122と、PSRAMコントローラ124と、IRQコントローラ126とを備えている。

【0039】コントロールレジスタ110は、各メモリのアドレスマッピングの変更や、IRQコントロール、ライトプロテクト等の指定を行う15種類のレジスタブロックを含む。CPU30から各レジスタへの書き込み時には、信号FE0～FE14の内の1つがアクティブになってレジスタブロックを指定し、同時に信号CD7Iにより書き込みデータが与えられ、信号CPUWRBにより書き込みパルスが与えられる（データは、8ビット中の1ビットD7だけが使用される）。各レジスタブロックからCPU30への読み出し時には、信号FE0～FE13でライト時と同様にレジスタを指定することにより、そのレジスタのデータがCD7Oとして出力される（信号FE14はリード不可）。アドレスデコーダ112から出力される信号Cは、信号FE0～FE13の内どれかがアクティブの状態ではイレベルになる。そのため、NANDゲート114、インバータ115は、信号Cがハイレベルで、かつ信号C PURDBがローレベルの時にスリーステートゲート116の出力を許可する。これにより、データCD7OはデータバスD7に出力され、CPU30に読み込まれる。コントロールレジスタ110の内部構成の詳細は、図11に示されている。

【0040】アドレスデコーダ112は、アドレスバスA上の信号をデコードすることにより、信号FE0～FE14を生成する。また、アドレスデコーダ112は、信号FE0～FE13のうちのどれかがアクティブの場合に、信号Cをハイレベルにする。さらに、アドレスデコーダ112は、各メモリが配置されるアドレスエリアをデコードし、その結果を信号ROMSELBとしてマップデコーダ120に与える。

【0041】マップデコーダ120は、コントロールレジスタ110からのコントロール信号に基づいて、CPU30による特定アドレスアクセス時に、信号CE1B～CE4Bのうちのどれかをアクティブにする。これによって、特定アドレスに指定のメモリ回路が割り当てられる。さらに、マップデコーダ120は、CIC78からのカートリッジ真偽判定出力R10に基づいて、フラッシュメモリ76へのライトプロテクト信号WPBを制御する。マップデコーダ120の内部構成の詳細は、図10に示されている。

【0042】アドレスセクタ122は、コントロールレジスタ110からの信号MAPMODEに基づいて、アドレスマップ上へのメモリ回路の割り当てパターンを制御する。このアドレスセクタ122により、20モードと21モードという2種類のマップモードを選択できる。20モードでは、各メモリバンクの0000H～

7FFFHと8000H～FFFFHに同じメモリイメージが配置される。ゲーム機本体においては、00H～3FH、80H～BFHバンクの0000H～7FFFHにワークRAMやPPU、APU制御ポートが割り当てられているため、連続的にメモリを割り当てると0000H～7FFFHの領域においてはメモリを読み出せないという不具合が生ずる。したがって、20モードに設定することによりメモリに無駄が生じない。しかし20モードでは、0000H～7FFFHと8000H～FFFFHに同じメモリイメージが配置されるため、実質上メモリ容量の上限がメモリマップエリアの1/2に制限される。したがって、大容量のメモリが必要な場合には、連続的にメモリを配置する方がよい。21モードは、メモリを連続的に配置するモードである。アドレスセクタ122は、これら2つのマップモードを切り替える。

【0043】PSRAMコントローラ124は、疑似SRAM74に対してリフレッシュやアウトプットイネーブル制御、ライトプロテクトの制御を行うものであり、その内部構成の詳細は、図9に示されている。

【0044】IRQコントローラ126は、プログラムによる割り込み許可信号ENIRQ、割り込み終了信号SRCIRQおよびフラッシュメモリからの書き込み完了信号RY/BYBに基づいて、システムクロックSYSCCKに同期した信号IRQBを生成し、CPU30に与える。信号IRQBは、同様にコントロールレジスタ110にも与えられ、信号SRCIRQの再リセットに使用される。IRQコントローラ126の内部構成の詳細は、図8に示されている。

【0045】図8は、図7におけるIRQコントローラ126の内部構成を示すブロック図である。図8において、このIRQコントローラ126は、Dフリップフロップ（以下、D-FFと称す）130と、インバータ135と、D-FF134と、ORゲート136とを備えている。

【0046】D-FF130は、フラッシュメモリ76の書き込み完了時に、信号RY/BYBがローレベルからハイレベルに変化したとき、この時点で信号ENIRQをラッチする。通常、このD-FF130は、ローレベルの状態を維持しているため、信号ENIRQがローレベルの場合には、たとえクロックが与えられたとしても状態は変化しない。従って、信号ENIRQがハイレベルの時のみ信号IRQBが発生される。

【0047】D-FF130の状態がローレベルからハイレベルに変化すると、D-FF134のD入力はハイレベルからローレベルに変化する。D-FF134は、D入力がローレベルになった直後のシステムクロック（CPUの動作クロック）SYSCCKの立ち上がりで、このローレベル信号をラッチする。これによって、信号IRQBがハイレベルからローレベルに変化し、CPU



30に割り込みがかかる。D-FF134は、このようにして、信号IRQBの出力タイミングを調整する。

【0048】ORゲート136は、信号IRQBが出力された後、プログラムによって信号SRCIRQがローレベルからハイレベルに変化させられると、ハイレベルを出力する。これによって、D-FF130がリセットされる。D-FF130の出力がローレベルになると、D-FF134のD入力ハイレベルとなり、次のSYSCKの立ち上がりで信号IRQBがハイレベルに戻る。信号IRQBがハイレベルになると、コントロールレジスタ110により、信号SRCIRQは自動的にハイレベルからローレベルに変化する。また、D-FF130とD-FF134は、リセット信号RESETBによってクリア（およびセット）され、初期化される。

【0049】図9は、図7におけるPSRAMコントローラ124の内部構成を示すブロック図である。図9において、このPSRAMコントローラ124は、D-FF140、142と、インバータ144と、ANDゲート146と、インバータ148と、反転入力のNANDゲート150と、反転入力のNORゲート152と、NANDゲート154と、インバータ156、158とを備えている。

【0050】D-FF140は、D-FF142と、インバータ144、148と、ANDゲート146と協働して、システムリセット時の疑似SRAM74に対するリフレッシュ信号を生成する。通常使用状態（リセット信号RESETBがハイレベル）の時、D-FF140、142は、常にハイレベルの状態にセットされ、リフレッシュ信号の生成を行わない。リセット状態（リセット信号RESETBがローレベル）の時は、信号CE1Bが必ずハイレベルになり、CE端子入力がハイレベルになって、D-FF140が能動化される。その後、システムクロックとは非同期に動作するCIC78のためのクロック信号CICKの最初の立ち上がりのパルスによってD-FF140にローレベルがセットされ、同様に最初の立ち下りのパルスによってD-FF142にローレベルがセットされる。これにより、D-FF142は、この後リセット信号が再びハイレベルになるまでリフレッシュ信号をローレベルのまま維持する。

【0051】NANDゲート150は、通常アクセス時の疑似SRAM74のアウトプットコントロールを行う。信号CE1Bがローレベルでかつ信号CPURDBがローレベルのときには、CPU30が疑似SRAM74をアクセスしている状態であり、この場合には、ローレベルを出力して疑似SRAM74からのデータ出力を許可する。

【0052】NORゲート152には、リセット時のリフレッシュ、通常時のアウトプットコントロールとは別に、通常時のリフレッシュ制御として、信号REFRESHBが入力される。負入力のNORゲート152は、

これら3種類の信号のどれか1つがアクティブになった場合、信号POEB/RFBを疑似SRAM74に与える。

【0053】NANDゲート154は、インバータ156および158と協働して、疑似SRAM74への書き込み制御信号PWEBを生成する。NANDゲート154は、信号SYSCKがハイレベルで、かつプログラムによる疑似SRAM74への書き込み許可信号ENRAMWRがハイレベルで、かつ疑似SRAM74のチップイネーブル信号CE1Bがローレベル（アクティブ）

で、かつCPU30からの書き込み制御信号CPUWRBがローレベル（アクティブ）の時に、疑似SRAM74への書き込み制御信号PWEBをアクティブにする。

【0054】図10は、図7におけるマップデコーダ120の内部構成を示すブロック図である。図10において、このマップデコーダ120は、ROMデコーダ160と、PS-RAMデコーダ162と、フラッシュデコーダ164と、メモリユニットデコーダ166と、ANDゲート168とを備えている。

【0055】ROMデコーダ160、PS-RAMデコーダ162、フラッシュデコーダ164およびメモリユニットデコーダ166は、アドレスバスAを介して与えられるアドレスをデコードすることにより、それぞれ、マスクROM72、疑似SRAM74、フラッシュメモリA76およびフラッシュメモリB106をイネーブル状態にするためのチップイネーブル信号CE2B、CE1B、CE3BおよびCE4Bを出力する。

【0056】ここで、メモリ割り当て時のマスクROM72、疑似SRAM74、フラッシュメモリA76およびフラッシュメモリB106の優先順位は、一例としてマスクROM>疑似SRAM>フラッシュメモリA>フラッシュメモリBに選ばれている。

【0057】マスクROMエリアと疑似SRAMエリアが重なり、ROMデコーダ160およびPS-RAMデコーダ162が同時にアクティブになるべき場合でも、信号CE2Bにより、疑似SRAMデコーダ162はアクティブ化を阻害される。そのため、信号CE1Bが非アクティブ、信号CE2Bがアクティブとなり、マスクROM72が優先的にアクセスされる。

【0058】同様に、信号CE2B、CE1Bのどちらかがアクティブの場合には、フラッシュデコーダ164のアクティブ化が阻害され、信号CE3Bはアクティブにならない。また、信号CE2B、CE1B、CE3Bのどれかがアクティブの場合には、メモリユニットデコーダ166のアクティブ化が阻害され、信号CE4Bはアクティブにならない。

【0059】ANDゲート168の出力信号WPBは、フラッシュメモリA76およびフラッシュメモリB106への書き込み禁止指示信号ENFMWRとCIC78からの真偽判定結果R10のどちらがローレベル（アク

タイプ) になっても、ローレベルとなり、フラッシュメモリへの書き込みが禁止される。

【0060】図11は、図7におけるコントロールレジスタ110の内部構成を示すブロック図である。図11において、このコントロールレジスタ110は、D-FF180と、インバータ181と、ORゲート182と、ANDゲート184と、D-FF186と、ANDゲート188と、レジスタブロック190～300と、ORゲート198とを備えている。

【0061】D-FF180は、通常、ローレベルの状態を保持しており、割り込みが入った(信号IRQBがローレベルになった)後に、プログラムによってこのレジスタ(信号FE0で指定される)に書き込みが行われた時のみ、ハイレベルの状態に移行する。D-FF180がハイレベル状態になると、IRQコントローラ126により、信号IRQBがハイレベルに戻される。これにより、D-FF180は再びローレベルになる。また、リセット信号RESETBがローレベル時(システムリセット時)にも、インバータ181、ORゲート182により、D-FF180はリセットされる。

【0062】CPU30が特定アドレスをアクセスすると、信号FE0がハイレベルになるため、ANDゲート184からは、信号IRQBの状態が信号RD0として出力される。この信号RD0は、ORゲート198、スリーステートゲート116(図7参照)を介して、CPU30に読み込まれる。D-FF186は、CPU30からこのレジスタ(信号FE1で指定される)に対して書き込みが行われた時、そのデータを保持し、割り込み許可信号ENIRQとして出力する。CPU30が特定アドレスをアクセスすると、信号FE1がハイレベルになるため、ANDゲート188からは、信号ENIRQの状態が信号RD1として出力される。この信号RD1は、ORゲート198、スリーステートゲート116を介して、CPU30に読み込まれる。

【0063】レジスタブロック190は、D-FF192, 194, ANDゲート196を含み、CPU30から書き込まれたマップモードを保持し、信号MAPMODEとして出力する。CPU30は、まず、D-FF192に対してマップモード設定データの書き込み信号FE2を発生し、次にD-FF194に対して書き込み信号FE14を発生する。信号FE14に応答して、D-FF192の状態がD-FF194にラッチされ、設定したマップモードが有効になる。このように2段階の設定手法をとる理由は、レジスタブロック190～レジスタブロック300への設定を順次行い、これらの設定を同時に有効にすることにより、アドレスマップ変更時の移行期間のマップ状態を考慮する必要がなくなり、プログラムの作成が容易になるという利点があるからである。また、ANDゲート196により、現在のマップモードの設定値をCPU30からリードできる。なお、他

のレジスタブロック200～300もレジスタブロック190と同様の回路構成を有している。すなわち、レジスタブロック200～300のそれぞれは、前段のD-FF202～302と、後段のD-FF204～304と、ANDゲート206～306とを含む。

【0064】CPU30が特定のレジスタブロックをリードすると、それに対応するレジスタブロックのみが、その出力(RD0～RD13)にロー/ハイレベルの2つの状態を持つことができる。CPU30により指定されなかったレジスタブロックは、ANDゲート184, 188, 196～306により、その出力がローレベルに固定される。各レジスタブロックの出力の論理和をとるORゲート198の出力信号CD70は、CPU30に指定されたレジスタブロックの出力がハイレベルの時にハイレベルになり、当該レジスタブロックの出力がローレベルの時にローレベルになる。すなわち、CPU30から指定されたレジスタブロックの内容が、信号CD70として出力されることになる。

【0065】図12は、図5におけるデータ放送デコーダ82の内部構成を示すブロック図である。図12において、このデータ放送デコーダ82は、アンプ/VXOコントローラ402と、プリデコーダ404と、チャンネルデテクタ406と、チャンネルデコーダ408と、ホストインターフェース410と、PCMデコーダインターフェース412と、EEPROMポート414と、拡張ポート416とを備えている。

【0066】アンプ/VXOコントローラ402は、ビットストリーム信号BSTをアンプで増幅し、信号BST02としてPCMデコーダ84へ出力する。また、アンプ/VXOコントローラ402は、PCMデコーダ84からのビットストリームクロック信号BSCKとビットストリーム信号BSTとの位相誤差を検出し、VXO90に信号P/Dとしてフィードバックすることにより、クロック再生用のPLL回路を構成する。

【0067】PCMデコーダ84は、信号BST02をデコードし、信号FDINとしてデータ放送デコーダ82に返す。プリデコーダ404は、信号FDINを受け、データチャンネルをデコードし、シリアルデータSDOとしてチャンネルデテクタ406に出力する。このプリデコーダ404は、システムシーケンサ404a、M・Sフレーム同期検出回路404bと、パケット斜めデインターリーブ回路404cと、BCH(16, 5)誤り訂正回路404dと、SDSC(272, 190)誤り訂正回路404eとを含む。

【0068】システムシーケンサ404aは、クロックMCKを基に各種タイミング信号を生成し、デコードシーケンスをコントロールする。また、システムシーケンサ404aは、信号FDINに同期したPCMデコーダ84からのビットストリームクロックBSCKに基づいて、シリアルデータSDOに同期したビットクロックB

TCKを生成し、チャンネルデクタ406に出力する。さらに、システムシーケンサ404aは、シリアルデータSDOにおけるパケットの先頭を示す信号PSYNCを生成し、チャンネルデクタ406に出力する。

【0069】M・Sフレーム同期検出回路404bは、信号FDINのフレーム同期信号に基づいて、マスターフレームやスーパーフレームの同期検出を行う。パケット斜めデインターリーブ回路404cは、放送データをパケット毎のデータ列に並べ変える（通常、放送データは、誤り訂正の効率を上げるために、斜めインターリーブされて送信される）。

【0070】BCH（16、5）誤り訂正回路404dは、デインターリーブされた放送データの各パケットのパケットヘッダ部の誤り訂正を行う。誤り訂正をしきれずにエラーが残った場合、BCH（16、5）誤り訂正回路404dは、信号HERRを出力して、ヘッダ部にエラーが発生したことを各ブロックに知らせる。

【0071】SDSC（272、190）誤り訂正回路404eは、パケットデータ本体の誤り訂正を行う。誤り訂正をしきれずにエラーが残った場合、SDSC（272、190）誤り訂正回路404eは、信号PERRを出力して、データ本体にエラーが発生したことを各ブロックに知らせる。

【0072】チャンネルデクタ406は、チャンネルデコーダ408と協働して、CPU30が指定した論理チャンネルの放送データを抜き出し、CRC演算を行い、バッファリングし、CPU30へ出力する。また、チャンネルデクタ406は、論理チャンネルの一致検出を行い、バッファリングの許可を与える。このチャンネルデクタ406の内部構成の詳細は、図13に示されている。

【0073】チャンネルデコーダ408は、CPU30が指定したチャンネルのデータをデコードし、バッファリングするための回路を2系統含む（チャンネル1デコーダ/バッファ408a、チャンネル2デコーダ/バッファ408b）。これらチャンネル1デコーダ/バッファ408a、チャンネル2デコーダ/バッファ408bは、同様の回路構成を有するが、その内部構成の詳細は、図14～図19に詳細に示されている。

【0074】ホストインターフェース410は、入力ポート410a、出力ポート410b、アドレスデコーダ410c、レジスタ410d等を含み、各ブロックとホストコンピュータ間のインターフェースを行う。

【0075】PCMデコーダインターフェース412は、PCMデコーダ84とCPU30間のインターフェースを行う入出力ポートを含む。EEPROMポート414は、図5のEEPROM88とCPU30との間のインターフェースを行う入出力ポートを含む。EEPROM88は、シリアル入出力タイプを採用し、これに与えるチップセレクトCS、シリアルクロックSK、出力

データDOおよび入力データDIが、全てプログラムによりコントロールされる。拡張ポート416は、例えばハードディスク等を接続するためのインターフェース用入出力ポートを含む。

【0076】図13は、図12におけるチャンネルデクタ406の内部構成の詳細を示すブロック図である。図13において、このチャンネルデクタ406は、S-P変換回路420と、ビットカウンタ422と、設定レジスタ424と、チャンネル1用のLCI比較回路（以下「第1の比較回路」という）426と、D-FF428、430、436、438と、チャンネル2用のLCI比較回路（以下「第2の比較回路」という）434と、ANDゲート432、440とを備えている。

【0077】S-P変換回路420は、プリデコード404からのシリアルデータSDOを8ビット単位のパラレルデータに変換する。ビットカウンタ422は、信号PSYNCによりクリアされ、信号BTCKによりアップカウントされるカウンタを含み、パケットデータをデコードするための各種信号を出力する。ビットカウンタ422から出力される各種信号のタイミングの詳細は、図21のタイムチャートに示されている。

【0078】設定レジスタ424は、CPU30から設定される論理チャンネルデータを記憶する。論理チャンネルデータは、5ビットのLCI1データと、6ビットのLCI2データとを含み、本実施例ではこれを2系統持つ。信号WE0、WE1、WE2、WE3は、それぞれ、チャンネル1のLCI1データ、チャンネル1のLCI2データ、チャンネル2のLCI1データ、チャンネル2のLCI2データの書き込み許可信号であり、CPU30から出力される信号PAWRに同期して書き込まれる。RE0～RE3は、設定データの読み出し許可信号である。設定レジスタ424に設定されたデータは、C1L1、C1L2、C2L1、C2L2データとして出力され、論理チャンネルデータの一一致判定に利用される。同時に、設定レジスタ424には、データグループ構成を指示する信号CH1DGSTY、CH2DGSTYがCPU30から書き込まれる。これら信号CH1DGSTY、CH2DGSTYは、それぞれ1ビットの信号であり、上記LCI1データと同時に書き込まれる。

【0079】次に、図21のタイムチャートを参照して、第1の比較回路426、D-FF428、430、ANDゲート432の動作を説明する。LCI1データはシリアルデータSDOの第1～第5ビットに対応し、LCI2データはシリアルデータSDOの第17～第22ビットに対応する。信号LCI1SYNCは、シリアルデータSDOの第5ビット目にハイレベルになる信号であり、第1の比較回路426およびD-FF428に与えられる。

【0080】第1の比較回路426は、信号LCI1S

YNCがハイレベルのときS-P変換420からのデータとC1L1データとを比較し、信号LCI1SYNCがローレベルのときS-P変換420からのデータとC1L2データとを比較する。第1の比較回路426は、比較の結果、両データが一致していればハイレベル信号を出力し、一致していなければローレベル信号を出力する。すなわち、シリアルデータSDOの第5ビット目のタイミングで論理チャンネルLCI1の一致/不一致が検査され、D-FF428に対してその結果が出力される。D-FF428は、信号LCI1SYNCがハイレベルでかつ信号BTCKの立ち上がりのタイミングで、第1の比較回路426の出力をラッチする。そのため、D-FF428は、結果的に論理チャンネルLCI1の比較結果を保持することになる。

【0081】次に、シリアルデータSDOの第22ビット目においては、S-P変換420からはLCI2データが出力される。このタイミングでは、信号LCI1SYNCはローレベルである。そのため、第1の比較回路426は、S-P変換420からのデータとC1L2データとを比較し、その結果をD-FF430に出力する。D-FF430は、シリアルデータSDOの第22ビット目でハイレベルとなるLCI2SYNC信号を受け、信号BTCKの立ち上がりで第1の比較回路426の出力をラッチする。そのため、D-FF430は、結果的に論理チャンネルLCI2データの比較結果を保持することになる。D-FF428のLCI1比較結果データと、D-FF430のLCI2比較結果データとは、ANDゲート432に出力される。ANDゲート432は、ヘッダエラーが無い場合（信号HERRがローレベルの時）、両比較結果データが「一致」を示していれば、バッファリング許可信号CH1ENAをハイレベルにする。

【0082】チャンネル2においても、上記と同様に、第2の比較回路434、D-FF436、438、ANDゲート440により、バッファリング許可信号CH2ENAが生成される。シリアルデータSDOの第30ビット目には、信号PFXSYNCがハイレベルになり、S-P変換420からは、プリフィックスデータ（SCC、CI、DTC）の計8ビットが出力される。このタイミングで信号BTCKが立ち上がると、D-FF442にはプリフィックスデータがラッチされる。ここで、SCCはスクランブルの有無を示す信号、CIはパケットの連続性を示すカウント値、DTCはデータグループの先頭と終端を示すフラグである。シリアルデータSDOの第288ビット目には、信号PERRLDがハイレベルになる。また、プリデコーダ404は、SDSC（272、190）誤り訂正の結果、訂正不能エラーが発生したかどうかのフラグをPERRとして出力する。このタイミングで信号BTCKが立ち上がると、D-FF444は、信号PERRをラッチする。

【0083】ビットカウンタ422は、これら以外にも、8ビット単位でデータのタイミングを示す信号DATASYNCと、実データの終端タイミングを示す信号DBLASTと、当該パケットのプリフィックスデータと、実データおよびフラグのすべてがそろったことを示す信号PERRSYNCとを発生する。

【0084】前述したように、本実施例では、デコーダ/バッファは2系統あるが（チャンネル1用のデコーダ/バッファ408aと、チャンネル2用のデコーダ/バッファ408b）、図14はそのうちのチャンネル1用のデコーダ/バッファ408aの内部構成を示している。なお、チャンネル2用のデコーダ/バッファ408bも、デコーダ/バッファ408aと同じ回路構造を持っているため、ここでは説明を省略する。図14において、このチャンネル1用のデコーダ/バッファ408aは、データコントローラ450と、プリフィックスメモリ452と、データメモリ454と、データセクタ456とを備えている。

【0085】データコントローラ450は、バッファリング許可信号CH1ENAを受け、データサイズのカウンタ、連続性指標CIのチェック、データグループ単位でのCRC演算を行い、バッファリングのためのアドレスを発生する。このデータコントローラ450の内部構成の詳細は、図15に示されている。プリフィックスメモリ452は、プリフィックスデータ記憶用バッファであり、データコントローラ内のプリフィックスFIFOインターフェース472（図15参照）により、FIFOメモリとして使用される。データメモリ454は、データグループデータ記憶用バッファであり、データコントローラ内のデータFIFOインターフェース474（図15参照）により、FIFOメモリとして使用される。データセクタ456は、CPU30からのアクセスに対応し、プリフィックスデータ、データグループデータ、バッファリングされたパケット数のうちのどれかをCPU30に出力する。また、データセクタ456は、内部にエラーフラグチェックのためのORループを含み、連続して転送した複数パケット分のエラーフラグにエラーがあったかどうかを簡単に判定することができる。このデータセクタ456の内部構成の詳細は、図16に示されている。

【0086】図15は、図14におけるデータコントローラ450の内部構成の詳細を示すブロック図である。図15において、このデータコントローラ450は、データグループシーケンサ460と、データサイズカウンタ462と、CIカウンタ464と、排他的論理和ゲート466と、CRC回路468と、ORゲート470と、プリフィックスFIFOインターフェース472と、データFIFOインターフェース474とを備えている。

【0087】データグループシーケンサ460は、タイ

ミング発生回路460aを含み、データサイズのカウン  
ト、CIのカウン、CRC演算、FIFOへのデータ  
の書き込み等のタイミングを制御するためのタイミング  
信号を発生する。また、データグループシーケンサ46  
0は、受信許可フラグレジスタ460bおよびクリアフ  
ラグレジスタ460cを含み、CPU30からの制御に  
従って、データ放送の受信許可または受信禁止ととも  
に、バッファクリアを行う。

【0088】以下に、各ブロックの説明を交え、データ  
グループシーケンサ460の動作を説明する。

#### 1) 受信許可および禁止

CPU30が特定の番地にデータを書き込むと、CPU  
30から信号PAWRが、ホストインターフェース41  
0内のアドレスデコード410cから信号WE10が与  
えられ、データバスB上のデータが許可フラグレジスタ  
460bに書き込まれる。許可フラグレジスタ460b  
の出力は、各ブロックへの制御信号の出力を許可し、こ  
れによりデータ放送受信が許可される。なお、制御信号  
の出力は、信号CH1ENAによっても制御される。す  
なわち、論理チャンネルが一致しない場合には、たとえ  
許可フラグがONになっていてもデータ放送は受信され  
ない。また、データの受信開始は、先頭パケットである  
ことを示すプリフィックスデータが入力されることによ  
りトリガされるため、データグループの途中のパケット  
から受信されることはない。CPU30が特定番地をリ  
ードすると、信号RE10がアドレスデコード410c  
から与えられ、許可フラグレジスタ460bの内容をリ  
ードできる。

【0089】CPU30が別の特定番地にデータを書き  
込むと、信号PAWR、WE11によりクリアフラグレ  
ジスタ460cにデータが書き込まれる。クリアフラグ  
レジスタ460cにデータが書き込まれた場合、その出  
力信号BFCLRは、各ブロックに対するリセット信号  
として作用する。その後、クリアフラグレジスタ460  
cは、自動的にクリアされる。また、システムリセット  
信号RESETBが与えられた場合には、クリアフラグ  
レジスタ460cにデータが書き込まれた時と同様に、  
信号BFCLRによるリセット動作を行う。

【0090】2) データサイズカウンタ462の制御  
データグループの構成は2種類あり、設定レジスタ42  
4(図13参照)からの信号CH1DGSTY(CPU  
30により設定される)により識別される。このうち、  
第1のデータグループ構成では、転送データのサイズを  
示すデータDGSが与えられる。ここで、データグルー  
プとは、前述のパケットが複数個連なって構成されるも  
ので、このデータグループの先頭パケットの最前部にデ  
ータグループヘッダが置かれ、その中にデータDGSが  
含まれる。データDGSは、先頭パケットの3~5バイ  
ト目を占めるため、このタイミングで信号DSLDDが発  
生し、データサイズカウンタ462に3バイト分のデー

タDGSがロードされる。

【0091】その後、データサイズカウンタ462に  
は、信号DATASYNCに基づいて信号DSEDCが  
与えられ、信号BTCKの立ち上がりでダウンカウン  
トカウントされる。これにより、データサイズカウンタ4  
62は、複数パケットにまたがってデータサイズをダウ  
ンカウントし続け、データの残りが1になった時点で信  
号DSEQを発生する。なお、データサイズカウンタ4  
62は、CRC演算を行うタイミングと、その結果をロ  
ードするタイミングとを決定するためのものであり、こ  
れによってデータのバッファリングを中止するものでは  
ない。すなわち、データのバッファリング中止は、プロ  
グラムにより上述のクリアフラグレジスタ460cをO  
Nにすることによって制御される。ハードウェア的に  
は、システムリセットRESETBが発生しない限り、  
自動的にバッファリングが中止されることはない。

#### 【0092】3) CRC回路468の制御

CRC回路468は、転送されたデータグループデータ  
にエラーがないかどうかをチェックするための回路であ  
り、本実施例ではこれを自動的に行うことにより、ソフ  
トウェアの負担を軽減している。CRC回路468は、  
本願出願人が出願人である特開平5-151007号公  
報に詳細に述べられる回路構造を基本とし、これを8ビ  
ットパラレルデータを入力とし、生成多項式を固定化し  
た回路が用いられる。データグループの先頭パケット  
において信号CRCCLRが発生すると、CRC演算の初  
期値が0に設定される。その後、信号DATASYNC  
に基づいて生成された信号CRCNXTに同期して、各  
データ毎にCRC演算が行われる。CRC演算において  
は、エラーが無い場合、演算結果が最終的に0になるた  
め、CRC回路468は、演算結果が0であるかどうか  
を判定する回路を含む。CRC演算の終了時には、CR  
C回路468に信号CRCOKLDが与えられ、CRC  
演算結果が0であればローレベルが、0でなければハイ  
レベルがラッチされる。

【0093】CRC演算の終了タイミングは、データグ  
ループ構成によって異なる。第1のデータグループ構成  
の場合は、データサイズカウンタ462からの信号DS  
EQに基づいて信号CRCOKLDが生成され、第2の  
データグループ構成の場合は、パケット終端を示すプリ  
フィックスデータと当該パケットのデータの終わりを示  
す信号DBLASTに基づいて信号CRCOKLDが生  
成される。

【0094】CRCの演算結果は、ORゲート470に  
出力される。このORゲート470は、CRC演算でエ  
ラーが発生したか、または訂正不能の誤りがあった場合  
(信号PERRLTがハイレベルの場合)には、信号CR  
CERRをハイレベルにして受信データに誤りがある  
ことを示す。

#### 【0095】4) CIカウンタ464の制御

CIカウンタ464は、データグループ内における各パケットの連続性を確認するための回路である。各パケットのプリフィックスデータには、4ビットのCIデータが含まれており、同一データグループの連続するパケット毎に0、1、2、…というように連番がふつてある（15の次は0にもどる）。信号CILDは、先頭パケットのプリフィックスデータのラッチタイミングを示す信号PFXSYNCがハイレベルになった次のビットタイミングで発生され、チャンネルデテクタ406内にラッチされたCIデータをCIカウンタ464にロードする。以降、信号PERRSYNCに同期してCIカウンタ464がアップカウントされる。CIカウンタ464のカウンタ値とパケットプリフィックスのCIデータは、排他的論理和ゲート466で比較され、不一致の場合には、信号CIERRがハイレベルとなりパケット連続性エラーを示す。具体的には、あるパケットにおいてヘッダーエラーが生じた場合（信号HERRがハイレベルの場合）には、チャンネルイネーブル信号（例えば、CH1ENA）がアクティブにならず、パケット抜けが生ずる。CIカウンタ464は、このようなパケット抜けを検出するために有効である。

【0096】5) プリフィックスFIFOインターフェース472の制御

プリフィックスFIFOインターフェース472は、図14のチャンネル1プリフィックスメモリ452に対するアドレス信号を生成し、かつ、蓄えたプリフィックスデータの数（すなわち、パケット数）をカウントする。このとき、データグループシーケンサ460は、メモリへの書き込み制御信号PFIFOWRを、信号PERRSYNCに基づいて生成する。プリフィックスメモリ452が一杯になった時には、信号PFULが出力され、それ以上のデータ書き込みが禁止される。このプリフィックスFIFOインターフェース472の内部構成の詳細は、図17に示されている。

【0097】6) データFIFOインターフェース474の制御

データFIFOインターフェース474は、図14のチャンネル1データメモリ454に対するアドレス信号を生成する。メモリへの書き込み制御信号DFIFOWRは、信号DATASYNCに基づいて生成される。データメモリ454が一杯になった時には、信号DFULが出力され、それ以上のデータ書き込みが禁止される。このデータFIFOインターフェース474の内部構成の詳細は、図18に示されている。

【0098】図16は、図14におけるデータセレクト456の内部構成の詳細を示すブロック図である。図16において、このデータセレクト456は、D-FF480、482と、マルチプレクサ484と、ORループ483とを備えている。ORループ483は、ORゲート488、494と、D-FF486と、インバータ4

90と、遅延回路492とを含む。

【0099】D-FF480は、通常動作時は、信号CLK19毎にデータをラッチし続けるが、プリフィックスメモリ452へのデータ書き込み時には、データをラッチしない。従って、D-FF480には、常にCPU30が次にリードすべきプリフィックスデータがラッチされる。D-FF482は、D-FF480と同様に、常にCPU30が次にリードすべきデータグループデータをラッチする。これらD-FF480、482の動作の詳細は、図22のタイミングチャートを用いて後述する。

【0100】CPU30が特定番地をリードすると、その時のアドレス値に従って、ホストインターフェース410内のアドレスデコーダ410cから、信号RE20、PFIFORD、DFIFORD、RE21が出力される。これらの信号RE20、PFIFORD、DFIFORD、RE21は、マルチプレクサ484に入力され、それぞれ、入力端子I0、I1、I2、I3から入力されるデータを、データバスBに対して出力するよう、マルチプレクサ484の内部状態を切り替える。

【0101】データPFCNT1は、現在バッファメモリ452、454にたまっているプリフィックスデータのカウンタ値（すなわち、受信したパケットの数）であり、信号RE20によって選択される。データC1PODは、プリフィックスメモリ452に蓄えられた、次にCPU30にリードされるべきプリフィックスデータであり、信号PFIFORDによって選択される。データC1DODは、データメモリ454に蓄えられた、次にCPU30にリードされるべきデータグループデータであり、信号DFIFORDによって選択される。入力端子I3には、ORループ483の出力が与えられ、信号RE21によって選択される。

【0102】プリフィックスデータには、CIエラーフラグ、CRCエラーフラグ（SDSC誤り訂正不能エラーを含む）が含まれているが、多数のパケットデータがバッファメモリ454内に蓄えられている場合に、これらのエラーフラグを逐一チェックするには、長時間を要する。そのため、どれか1つでもエラーがあったかどうかをチェックするために、プリフィックスデータの時間軸方向に対するORが検出できれば、それだけCPU30の負担が減少する。ORループ483は、このような目的のために使用される。

【0103】D-FF486は、バッファクリアを指示する信号BFCLRにより、ORゲート494を介してクリアされる。また、一旦、信号RE21がアクティブになり、D-FF486のデータがCPU30にリードされた直後にも、遅延回路492、ORゲート494を介してD-FF486がクリアされる。この状態で信号PFIFORDがアクティブになり、プリフィックスデータがデータバスBに出力されると、ORゲート488

は、D-FF486の出力とプリフィックスデータとのORをとってD-FF486に与える。CPU30にプリフィックスデータが読み込まれ、信号PFIFORDが立ち下がる時に、D-FF486はORゲート488からのORデータをラッチする。ここで、さきほどのプリフィックスデータにエラーがあれば、そのビットがハイレベルになるが、エラーがなければそのビットはローレベルのままである。こうしてCPU30が順次プリフィックスデータをリードすることにより、1つでもエラーがあればD-FF486にハイレベルをラッチするよう、ORループを構成する。すべてのプリフィックスデータをリードし終わると、CPU30は、D-FF486の内容をリードし、エラーがあったかどうかをチェックする。

【0104】図17は、図15におけるプリフィックスFIFOインターフェース472の内部構成の詳細を示すブロック図である。図17において、このプリフィックスFIFOインターフェース472は、タイミング発生回路500と、プリフィックスラッチ502と、ライト用ループカウンタ504と、リード用ループカウンタ506と、インバータ508と、ORゲート510と、アドレスマルチプレクサ512と、プリフィックスデータ数カウンタ514と、デコーダ516と、インバータ518とを備えている。

【0105】放送データの書き込みタイミングは、CPU30の動作クロックSYSCKとは非同期に発生するため、タイミングによってはCPU30からのリードと同時に発生する場合がある。タイミング発生回路500は、データのリードまたはデータ数カウント値のリードと、データの書き込みまたはデータ数のアップカウント（または、ダウンカウント）とが同時に発生しないように、タイミングを調整する。基本的には、信号SYSCKがハイレベルの期間にデータリードまたはデータ数カウント値のリードが、信号SYSCKがローレベルの期間にデータのライトまたはアップカウント（およびダウンカウント）が発生するよう調整される。なお、D-FF480でリードデータをラッチしているため、データの書き込みタイミングが信号SYSCKのハイレベル期間にまでずれ込んでも問題は生じない。これによって、FIFOのライトサイクルを制御するクロックCLK19に9MHzという比較的遅いクロックを使用することができる。

【0106】タイミング発生回路500は、データグループシーケンサ460からの書き込み制御信号PFIFOWRに基づいて、信号SYSCK、CLK19に同期した書き込み信号PWRENを生成する。プリフィックスメモリ452が満杯であることを示す信号PFULがアクティブの時には、信号PWRENは出力されない。また、タイミング発生回路500は、CPU30が特定番地をリードしたことを示す信号PFIFORDに基づ

いて、信号CLK19に同期した読み出し信号PRDENを生成する。プリフィックスメモリ452が空であることを示す信号PEMPがアクティブの場合には、信号PRDENは出力されない。さらに、タイミング信号500は、プリフィックスメモリ452へのデータライト時に発生するデータ数のアップカウントと同様のタイミングで、リード時のダウンカウントが発生するように、信号PDWを、読み出し制御信号DFIFORDおよびクロックSYSCKの立ち下がりに基づいて生成する。タイミング発生回路500の動作は、図22、図23のタイミングチャートを参照して、さらに詳細に述べられる。

【0107】プリフィックスラッチ502は、プリフィックスデータをビットクロックBTCKに同期してラッチする。ここでラッチされるデータは、スクランブル情報SCC（2ビット）、先頭・終端パケット情報DTC（2ビット）のみであるが、これにパケット連続性エラーフラグCIERR、CRC演算結果（SDSCエラー訂正結果を含む）CRCERRを加えた計6ビットが、プリフィックスデータとしてプリフィックスメモリ452に与えられる。

【0108】ライト用ループカウンタ504は、ライト時にプリフィックスメモリ452に与えるアドレス値を記憶する。また、ライト用ループカウンタ504は、データ書き込み信号PWRENが発生する毎に、1ずつアップカウントを行い、そのカウント値が最大値に到達すると、自動的に0に戻る。

【0109】リード用ループカウンタ506は、リード時にプリフィックスメモリ452に与えるアドレス値を記憶する。リード用ループカウンタ506は、データ読み出し信号PRDENが発生する毎に、1ずつアップカウントを行い、そのカウント値が最大値に到達すると、自動的に0に戻る。

【0110】信号PWRENがハイレベルになると、インバータ508は、ローアクティブのプリフィックスデータライトパルスC1PWEを、プリフィックスメモリ452に与える。信号PWRENが立ち上がってから信号C1PWEが立ち上がるまでの間、ORゲート510は、アドレスマルチプレクサ512にハイレベルのセレクト信号SELを与える。これによって、アドレスマルチプレクサ512は、書き込み期間の間だけ、ライト用ループカウンタ504からのアドレスデータを、アドレスC1PADRとしてプリフィックスメモリ452に出力する。信号SELがローレベルの間、アドレスマルチプレクサ512は、リード用ループカウンタ506からのアドレスデータを、アドレスC1PADRとしてプリフィックスメモリ452に出力する。

【0111】プリフィックスデータ数カウンタ514は、7ビットの加算器と7ビットのD-FFとにより構成されたアップダウンカウンタであり、+1および+7

FH(−1)を行うことにより、プリフィックスメモリ452にバッファリングされたプリフィックスデータの数(受信したパケットの数)をカウントする。

【0112】デコーダ516は、プリフィックスデータ数カウンタ514からのカウント値が0の時に、信号PEMPをハイレベルにし、当該カウント値がプリフィックスメモリ452の記憶容量と同じになった時(本実施例では47H)に、信号PFULをハイレベルにする。

【0113】インバータ518は、信号PWRENがハイレベルの時(データ書き込み時)に、DFF480(図16参照)によるデータラッチを禁止する。DFF480は、通常時はリードデータをラッチし続けるが、データ書き込み時には、インバータ518からの信号PFLATG1により、直前にラッチしたリードデータを保持する。

【0114】図18は、図15におけるデータFIFOインターフェース474の内部構成の詳細を示すブロック図である。図18において、このデータFIFOインターフェース474は、大部分は、図17のプリフィックスFIFOインターフェース472と同様の回路構成を有している。すなわち、タイミング発生回路520、データラッチ522、ループカウンタ524、526、インバータ528、540、ORゲート530、アドレスマルチプレクサ532、インバータ540は、それぞれ、図17のタイミング発生回路500、プリフィックスラッチ502、ループカウンタ504、506、インバータ508、518、ORゲート510、アドレスマルチプレクサ512に相当しており、その説明を省略する。データFIFOインターフェース474は、図17のプリフィックスFIFOインターフェース472と異なる構成として、DFF534、536と、EMP/FULデテクタ538とを備えている。

【0115】DFF534および536は、それぞれ、信号DWRENおよびDRDENを、信号CLK19の立ち上がりでラッチし、信号WCHKFFおよびRCHKFFとして、EMP/FULデテクタ538に出力する。

【0116】プリフィックスデータは、1パケットに1バイトであり、プリフィックスメモリ452の最大値は、47H(71)バイトである。これに対して、放送データは、例えば1パケットにつき22バイトのデータを含み、データメモリ454は、 $22 \times 71 = 1562$ バイトの記憶容量を持つ。しかし、プリフィックスメモリ452のデータと、データメモリ454のデータとは同時に読み出せないため、前述のデータ数カウンタ514だけではデータメモリ454のFULL、EMPTYは判断できない。したがって、データFIFOインターフェース474は、データメモリ454のFULL、EMPTYを判断するために、独自のEMP/FULデテクタ538を持つ。

【0117】図19は、図18におけるEMP/FULデテクタ538の内部構成の詳細を示すブロック図である。図19において、このEMP/FULデテクタ538は、一致判定回路550と、加算データデコーダ552と、加算器/ラッチ554と、E/Fデコーダ556とを備えている。

【0118】一致判定回路550は、ライト用ループカウンタ524からのライトアドレスWCと、リード用ループカウンタ538からのリードアドレスRCとの一致を判定し、一致を判定した場合には、ハイレベルの信号EQを出力する。加算データデコーダ552は、図20に示すように、各入力条件に対して所定の加算データを出力する。加算器/ラッチ554は、加算データデコーダ552からの加算データと、前回の演算結果とを加算し、信号CLK19の立ち上がりで加算結果をラッチする。E/Fデコーダ556は、加算器/ラッチ554にラッチされた加算結果に基づいて、信号DEMP、DFULを生成する。

【0119】次に、図18および図19に示すEMP/FULデテクタ538の動作を説明する。まず、信号BFCCLRにより、加算器554内のラッチは00(2進数2桁)になる(初期状態)。この時には、E/Fデコーダ556は、00をデコードし、信号DEMPにハイレベルを出力し、データメモリ454が空であることを示す。この状態で信号WCHKFFがハイレベルになると、

$$WCHKFF * EQ * DEMP * /RCHKFF = 1$$

の条件式(なお、\*はAND条件を、/はNOTを表している)が成立し、加算器/ラッチ554には加算データ01が与えられる。そのため、加算器/ラッチ554において、前回の加算結果00と、今回の加算データ01との加算結果が01となり、このデータが加算器/ラッチ554内に新たにラッチされる。E/Fデコーダ556は、加算器/ラッチ554のラッチデータ01をデコードして、信号DEMPをローレベルに下げ、信号RDYをハイレベルに上げる。この直後に、ライトアドレスWCの値とリードアドレスRCの値とが一致しなくなるため、その後はEQ=HIGHの条件が成り立たなくなる。すると、加算データは00となり、加算結果に変化は生じない。

【0120】さらに、データライト動作が続き、データメモリ454が満杯になると、再びライトアドレスWCとリードアドレスRCは一致する。この時には、 $WCHKFF * EQ * RDY * /RCHKFF = 1$ の条件式が成立するため、加算データ01が加算器/ラッチ554に与えられる。このとき、加算器/ラッチ554において、 $01 + 01 = 10$ が演算され、ラッチされる。そのため、E/Fデコーダ556において、信号RDYはローレベルに落ち、信号DFULがハイレベルとなる。信号DFULがハイレベルになると、これ以上



のデータの書き込みは禁止される。

【0121】次に、データリード動作が発生すると、信号RCHKFFがハイレベルになるため、 $RCHKFF * EQ * DFUL * \neg WCHKFF = 1$ の条件式が成立し、加算データ11が加算器／ラッチ554に与えられる。このとき、加算器／ラッチ554において、 $10 + 11 = (1)01$ となり、E/Fデコーダ556において、信号RDYが再びハイレベルになり、信号DFULはローレベルになる。このままデータリード動作を続けると、最終的には、 $RCHKFF * EQ * RDY * \neg WCHKFF = 1$ の条件が成立して、加算データ11が加算器／ラッチ554に与えられ、加算結果が $01 + 11 = (1)00$ となって、EMPTYの状態に戻る。このようにして、EMP/FULデテクタ538は、信号DEMP、DFULを生成する。

【0122】次に、図22のタイムチャートを参照して、データメモリ454（図14参照）のライトタイミングを説明する。信号SYSCCKは、例えば2.68MHzのクロック信号であり、CPU30の動作クロックとして用いられる。信号CLK19は、VXO90（図5参照）からのクロック信号MCKを2分周した9.216MHzのクロック信号であり、データメモリ454のリード／ライトクロックとして用いられる。信号BTCKは、2.048MHzのクロック信号であり、PCMデコーダ84（図5参照）からの信号FDINの同期クロックとして用いられる。クロック信号MCKの周波数は、VXO19により、例えばこのクロック信号BTCKの9倍の周波数18.432MHzに調整されている。

【0123】図15のデータグループシーケンサ460が、信号DATASYNCに基づいて、信号DFIFOWRを出力すると、信号BTCKの立ち上がりで、図18のデータラッチ522にデータグループデータがラッチされる。信号DFIFOWRがハイレベルになった後、信号SYSCCKが立ち下がってから2つ目の信号CLK19の立ち上がり（信号SYSCCKの立ち下がりよりT1の後）に、信号DWRENがハイレベルになり、信号SELがハイレベルになってアドレスマルチプレクサ532がライト用アドレスADDWRをデータメモリ454に出力する。

【0124】信号SYSCCKが立ち下がってから3つ目の信号CLK19の立ち上がりによって、信号DWRENはローレベルになり、信号DWRENの反転信号C1DWEの立ち上がりにより、データメモリ454の番地ADDWRにデータラッチ522のデータが書き込まれる。インバータ528の影響で信号DWRENの立ち下がりよりも信号C1DWEの立ち上がりが遅れる。そのため、信号C1DWEが立ち上がるまでアドレスマルチプレクサ532がライト用アドレスをセレクトするよ

う、ORゲート530によって信号SELを遅延させる。3つ目の信号CLK19の立ち上がりでさらにライト用ループカウンタ524がアップカウントされるが、当該ループカウンタ524からの信号出力“ADDWR+1”は、インバータ528による遅延よりもさらに遅れるため、データメモリ454に与えるアドレス信号には影響が生じない。

【0125】なお、プリフィックスメモリ452へのライトも上記と同様のタイミングで動作するが、プリフィックスデータ数カウンタ514は、やはり3つ目の信号CLK19の立ち上がりでカウントアップされる。

【0126】図22におけるその他の信号は、受信データ数が1562バイトに達した場合のEMP/FULデテクタ538の動作を示している。図22を参照して、D-FF534は、信号DWRENを信号CLK19の1クロック分遅らせた信号WCHKFFを発生する。一致判定回路550は、信号EQをハイレベルにする。信号RDYは、ハイレベルであったため、加算データには01が発生し、信号CLK19の4つ目のクロックで加算データ10がラッチされる。これによって、信号RDYがローレベルになり、信号DFULがハイレベルになる。

【0127】信号RDLATG1は、信号DWRENと同じタイミングでローレベルになり、信号CLK19の3つ目でのD-FF482によるデータラッチを禁止する。D-FF482には、信号CLK19の1つ目、2つ目、4つ目の立ち上がりで常にリードアドレスで示されるデータがラッチされており、CPU30からのデータリードを保証する。プリフィックスデータライトの場合も同様にD-FF480がCPU30からのプリフィックスデータリードを保証する。

【0128】次に、図23のタイムチャートを参照して、データメモリ454のリードタイミングを説明する。CPU30によるデータリードが発生すると、信号SYSCCKの立ち上がりから1つめの信号CLK19の立ち上がりで、信号DRDENがハイレベルになる。これに伴い、信号CLK19の2つ目のクロックでリード用ループカウンタ526の値がインクリメントされるが、この2つ目のクロックでは、D-FF482にはまだインクリメントされる前のアドレスによって示されるデータがラッチされている。インクリメント後のアドレスで示されるデータがD-FF482にラッチされるタイミングは、信号CLK19の3つ目の立ち上がりであるが、周波数を考慮するとその前に必ずCPU30によるリード動作が発生する（信号CPURDの立ち上がりがリード動作のタイミング）ため、リードアドレスのインクリメントがCPU30のリード動作に影響を及ぼすことはない。

【0129】図23におけるその他の信号は、バッファ内のデータ数が0になった場合のEMP/FULデテク

タ538の動作を示している。D-FF536は、信号DRDENを信号CLK19の1クロック分遅らせた信号RCHKFFを発生する。一致判定回路550は、信号EQをハイレベルにする。信号RDYは、ハイレベルであったため、加算データには11が発生し、信号CLK19の3つ目のクロックで加算データ00がラッチされる。これによって、信号RDYがローレベルになり、信号DEMPがハイレベルになる。

【0130】なお、プリフィックスメモリ452のリード時にも上記と同様のタイミングで動作するが、信号PDWはデータリードがあった後、信号SYSCCKの立ち下がりから2つ目の信号CLK19の立ち上がり（リード時の信号SYSCCKの立ち上がりから数えると4つ目の信号CLK19の立ち上がり）でハイレベルになり、その次の信号CLK19の立ち上がりでデータ数がデクリメントされる。このデクリメントのタイミングは、ライト時のインクリメントのタイミングと同じであるため、信号SYSCCKに同期してデータ数が確定する。

【0131】CPU30によるデータリードは、信号SYSCCKがハイレベルの時に発生するが、次のCPUリードタイミングまでにはすべてのインクリメント、デクリメント動作は終了し、安定化するため、DMAで連続的にデータをリードした場合でも、問題無く動作する。

【0132】次に、CPU30が管理するアドレス空間について説明する。図24は、CPU30が保有するアドレス空間に対応するメモリマップを示している。CPU30は、このメモリマップ上の斜線部の領域に、マスクROM72、疑似SRAM74、フラッシュメモリA76、フラッシュメモリB106を割り付けることにより、各メモリのアクセスを管理する。すなわち、CPU30は、限られた1つのアドレス空間内で、複数のメモリを一元的に管理することができる。なお、図24において、斜線部以外の領域は、CPU30の制御領域として固定的に用いられ、この部分においてワークRAM32等のアクセスが行われる。

【0133】通常、各メモリの総容量は、CPU30のアドレス空間の容量を越えているため、同一の部分に複数のメモリが重なって割り付けられる事態が生じる。本実施例では、各メモリにアクセスの優先順位を与えることにより、このような事態に対処している。すなわち、メモリマップの同一部分に複数のメモリが割り付けられている場合、優先順位の高いメモリの方が、優先順位の低いメモリよりも優先的にアクセスされる。本実施例では、一例として各メモリの優先順位を、優先順位の高い順からマスクROM>疑似SRAM>フラッシュメモリA>フラッシュメモリBと決めている。優先順位の高いメモリと低いメモリとを並列的にアクセスしたい場合は、いずれかのメモリの割り付け位置がメモリマップ上で移動される。

【0134】図25は、CPU30のメモリマップ上に

において、マスクROM72を割り付け可能な領域を示している。図25において、マスクROM72は、領域#Aおよび/または#Bに割り付けることができる。なお、領域#Aと領域#Bは、同一のマスクROMのイメージを示している。

【0135】図26は、CPU30のメモリマップ上において、疑似SRAM74を割り付け可能な領域を示している。なお、図26(a)は20モードにおける割付可能領域を示しており、図26(b)は21モードにおける割付可能領域を示している。図26(a)において、疑似SRAM74は、領域#A～#Hのいずれか1つまたは複数に割り付けることができる。また、後述する信号RAMONがハイレベルのとき、疑似SRAM74は、領域\*に常に割り付けられている。なお、領域#C'、#D'、#G'、#H'、##'には、領域#C、#D、#G、#H、##のイメージが現れている。図26(b)において、疑似SRAM74は、領域#A～#Hのいずれか1つまたは複数に割り付けることができる。また、後述する信号RAMONがハイレベルのとき、領域##には、領域#A～#Hの6000H～7FFFHのイメージが常に現れている。また、領域#A'～#H'には、領域#A～#Hの8000H～FFFFHのイメージが常に現れている。

【0136】図27は、CPU30のメモリマップ上において、フラッシュメモリA76を割り付け可能な領域を示している。なお、図27(a)は20モードにおける割付可能領域を示しており、図27(b)は21モードにおける割付可能領域を示している。図27(a)において、フラッシュメモリA76は、領域#A～#Dのいずれか1つまたは複数に割り付けることができる。また、領域#B'、#D'には、領域#B、#Dのイメージが現れている。図27(b)において、領域#A～#Dのいずれか1つまたは複数に割り付けることができる。また、領域#A'～#D'には、領域#A～#Dの8000H～FFFFHのイメージが現れている。

【0137】図28は、CPU30のメモリマップ上において、フラッシュメモリB106を割り付け可能な領域を示している。図28を図24と対比すれば明らかのごとく、フラッシュメモリB106は、メモリマップ上で各メモリを割付可能な領域の全てに割り付けることが可能である。

【0138】図29は、図5に示す衛星データ放送受信アダプタ4における信号の流れを示している。また、図30は、本実施例の衛星データ放送受信システムで採用されるデータ伝送の階層モデルを示している。以下、これら図29および図30を参照して、本実施例の全体的な動作を説明する。

【0139】(1)リセット時の動作

図3のパワースイッチ62がオンされると、電源回路64が動作を開始し、システム内の各部の電圧が上昇して

いく。これに伴って、リセット回路60内の所定のノードの電圧が上昇し、当該ノードが一定電圧以上になると、リセット回路60はリセット信号RESETBを一定時間アクティブ（この場合は、ローレベル）にする。また、リセット回路60は、リセットスイッチ58がオンされた場合にも、リセット信号RESETBを一定時間アクティブにする。このリセット信号RESETBは、コネクタA38を介して、衛星データ放送受信カートリッジ6内のマッピングコントローラ70（図4参照）に与えられる。

【0140】マッピングコントローラ70において、リセット信号RESETBは、図7のコントロールレジスタ110に与えられる。このコントロールレジスタ110内において、リセット信号RESETBは、図11のインバータ181でハイレベル信号に反転された後、各レジスタブロック190～300における後段のDFF194～304のクリア端子またはセット端子に与えられる。ここで、インバータ181の出力が、各レジスタブロックのDFF194～304のクリア端子に与えられるか、セット端子に与えられるかは、初期設定時における各メモリの配置をどうするかに応じて定められている。インバータ181の出力がクリア端子に与えられた場合、該当するDFFは、論理0を保持し、そのQ出力はローレベルとなる。これに対し、インバータ181の出力がセット端子に与えられた場合、該当するDFFは、論理1を保持し、そのQ出力はハイレベルとなる。

【0141】ここで、レジスタブロック190～300の内、少なくともレジスタブロック240および250については、インバータ181の出力がDFF244、254のセット端子に与えられる。これは、図4のマスクROM72にCPU30の動作プログラムが格納されているため、CPU30が起動する前にマスクROM72を読み出し可能状態、すなわちメモリマップ上に配置しておく必要があるためである。もし、初期状態においてマスクROM72がメモリマップ上に配置されていないと、動作プログラムを読み出せないため、CPU30は起動できなくなる。この初期設定によって、少なくともコントロール信号ROMON0およびROMON1がハイレベルとなる。このハイレベルのコントロール信号ROMON0およびROMON1は、コントロールバスを介して、マップデコーダ120内のROMデコーダ160（図10参照）に与えられる。これによって、マスクROM72は、図25のメモリマップ上の領域#Aおよび#Bに配置されたことになる。

【0142】（2）CPU起動時の動作  
上記リセットによるメモリの初期配置が終了すると、CPU30が起動する。起動したCPU30は、マスクROM72から衛星データ放送受信の動作プログラムを読み出すために、マスクROM72に対応するアドレスデ

ータをアドレスバスAに出力する。当該アドレスデータは、マッピングコントローラ70内のアドレスデコーダ112（図7参照）に与えられる。このアドレスデコーダ112は、与えられたアドレスデータがメモリマップ上の図24の斜線部で示す領域を指定しているとき、ローレベルの信号ROMSELBを出力する。また、当該アドレスデータは、マップデコーダ120内のROMデコーダ160に与えられる。このとき、ROMデコーダ160に与えられる各信号の論理状態は、以下のようになっている。

10 ROMON0=1  
ROMON1=1  
ROMSELB=0  
RESETB=1

【0143】図31は、ROMデコーダ160において、マスクROM72のチップイネーブル信号CE2Bがアクティブ（ローレベル）になる条件を示している。この図31において、A21～A23は、CPU30からのアドレスデータ（A0～A23）の22～24ビット目の値を示している。信号ROMSELBが論理0で、かつアドレスデータの23および24ビット目がいずれも論理0の場合は、アドレスデータが図25における領域#Aのいずれかの部分を示していることになる。また、信号ROMSELBが論理0で、かつアドレスデータの23ビット目が論理0で、24ビット目が論理1の場合は、アドレスデータが図25における領域#Bのいずれかの部分を示していることになる。

【0144】ROMデコーダ160に与えられる各信号の論理状態は、前述のようになっているため、ROMデコーダ160は、図31の条件に基づき、CPU30からのアドレスデータが現在のマスクROM72のメモリ配置にヒットしていることを判断し、マスクROM72を能動化させるためのチップイネーブル信号CE2Bをアクティブ（ローレベル）にする。これによって、マスクROM72の動作が許可され、CPU30は、マスクROM72をアクセス可能となる。

【0145】（3）マスクROMの再配置動作  
CPU30は、マスクROM72から衛星データ放送受信の動作プログラムを読み出すと、当該動作プログラムに従って、メモリマップ上でのマスクROM72の再配置すなわち割り付け位置の変更を行う。メモリマップ上では、現在、図25の領域#Aおよび#Bの両方にマスクROM72が割り付けられている。これでは使い勝手が悪く、メモリマップの使用効率も落ちるため、CPU30は、マスクROM72を領域#Aおよび#Bのいずれか一方に割り付ける。本実施例では、マスクROM72を領域#Aに割り付けている。

【0146】（4）メニューデータの受信動作  
次に、CPU30は、TVモニタ22上に初期画面（メーカー名や本システムのタイトル等）を表示し、現在放

送中の番組メニューを入手するために、TCDと呼ばれる番組一覧データ（メニューデータ）を受信する。この受信動作の詳細を以下に説明する。

#### 【0147】A. 論理チャンネルの設定

まず、CPU30は、マスクROM72から読み出したプログラムデータに従って、TCDのために予め定められている論理チャンネル（例えば、LCI1=2、LCI2=1）をデータ放送デコーダ82にセットする（図29のα1）。すなわち、CPU30は、チャンネル1 LCI1のI/OポートアドレスをアドレスバスBに出力すると共に、データバスに論理チャンネルLCI1のデータ「2」を出力し、制御信号Bに含まれる信号PAWRをローレベルにセットする。データ放送デコーダ82は、バスバッファ94、96、双方向バッファ92を介して、これらの信号を入力する。データ放送デコーダ82内のホストインターフェース410（図12参照）は、内部のアドレスデコーダ410cによって、アドレスバスB上のチャンネル1 LCI1のI/Oポートアドレスをデコードし、デコード信号C1L1LDを生成する。このデコード信号C1L1LDは、制御信号バスCCを介して、チャンネルデテクタ406に出力される。チャンネルデテクタ406内の設定レジスタ424（図13参照）は、デコード信号C1L1LDおよび制御信号PAWRを受けて、データバス上のデータ「2」を内部レジスタにストアする。これによって、設定レジスタ424からチャンネル1 LCI比較回路426に、論理チャンネルLCI1を指定するデータ「2」が、信号線C1L1を介して入力され、放送によって送られてくるデータパケットのデータLCI1と比較される。

【0148】次に、CPU30は、チャンネル1 LCI2のI/OポートアドレスをアドレスバスBに出力すると共に、データバスに論理チャンネルLCI2のデータ「1」を出力し、制御信号Bに含まれる信号PAWRをローレベルにセットする。データ放送デコーダ82内のホストインターフェース410は、論理チャンネルLCI1のときと同様に、内部のアドレスデコーダ410cによってアドレスバスB上のチャンネル1 LCI2のI/Oポートアドレスをデコードし、デコード信号C1L2LDを生成する。チャンネルデテクタ406内の設定レジスタ424は、上記デコード信号C1L2LDおよび制御信号PAWRを受けて、データバス上のデータ「1」を内部レジスタにストアし、信号線C1L2を介して、チャンネル1 LCI比較回路426に出力する。チャンネル1 LCI比較回路426では、設定レジスタ424から与えられたデータ「1」と、放送によって送られてくるデータパケットのデータLCI2とを比較する。

【0149】B. バッファメモリの初期化と受信許可の付与

次に、CPU30は、チャンネル1のクリアフラグレジ

スタ460c（図15参照）を示すI/OポートアドレスをアドレスバスBに出力すると共に、データバスにデータ「1」を出力し、制御信号Bに含まれる信号PAWRをローレベルにセットする。データ放送デコーダ82内のホストインターフェース410は、内部のアドレスデコーダ410cによってアドレスバスB上のクリアフラグレジスタアドレスをデコードし、デコード信号WE11を生成する。このデコード信号WE11は、制御信号バスCCを介して、チャンネルデコーダ408のデータコントローラ450内にあるデータグループシーケンサ460に出力される。

【0150】データグループシーケンサ460内のクリアフラグレジスタ460cは、上記デコード信号WE11および制御信号PAWRを受けて、データバス上のデータ「1」を記憶する。これによって、データグループシーケンサ460は、内部の受信許可フラグレジスタ460bを受信禁止の状態に設定し、信号BFCLRを出力してチャンネル1用バッファメモリ452、454（図14参照）のクリアを行う。ただし、バッファメモリ452、454は、FIFOとして構成されているため、実際にデータが消去されるのではなく、データが空の状態になるようにアドレス操作が行われることにより、データの消去が実現される。この信号BFCLRは、システムリセット時にも同様に発生する。データ「1」が書き込まれたクリアフラグレジスタ460cは、バッファクリア後、自動的に0に戻る。

【0151】次に、CPU30は、チャンネル1の受信許可フラグレジスタ460bを示すI/OポートアドレスをアドレスバスBに出力すると共に、データバスにデータ「1」を出力し、制御信号Bに含まれる信号PAWRをローレベルにセットする。データ放送デコーダ82内のホストインターフェース410は、内部のアドレスデコーダ410cによってアドレスバスB上の受信許可フラグレジスタアドレスをデコードし、デコード信号WE10を生成する。このデコード信号WE10は、制御信号バスCCを介して、チャンネルデコーダ408のデータコントローラ450内にあるデータグループシーケンサ460に出力される。データグループシーケンサ460内の許可フラグレジスタ460bは、上記デコード信号WE10および制御信号PAWRを受けて、データバス上のデータ「1」を記憶する。これによって、データグループシーケンサ460は、先頭パケット検出状態に移行する。

#### 【0152】C. パケットの生成

データ放送デコーダ82にビットストリーム信号BSTが入力されると、データ放送デコーダ82内のアンプ/VXOコントローラ402は、当該ビットストリーム信号BSTを増幅し、信号BST02としてPCMデコーダ84に出力する（図29のα2）。PCMデコーダ84は、与えられた信号BST02に対してビットデイン

ターリーブを施した後、音声信号とデータとに分離する。分離された音声信号は、PCMデコーダ84でPCM復調された後、アンプ86a、86b、ミキサ50a、50b、アンプ52a、52bを介して、TVモニタ22に出力される。これによって、本システムは、通常の音楽番組を受信するために使用可能である。一方、分離されたデータは、PCMデコーダ84で信号FDINとしてデータ放送デコーダ82に戻される(図29のα3)。データ放送デコーダ82内において、プリデコーダ404は、信号FDINを、斜めデインターリーブ、BCH誤り訂正、SDSC誤り訂正等を施した後、パケット単位の信号SDOとして、ビットシリアルに、チャンネルデテクタ406に出力する(図29のα4)。

【0153】上記信号SDOは、前述の図21に示すようなフォーマットを有している。チャンネルデテクタ406内のビットカウンタ422は、プリデコーダ404からのパケットの先頭を示す信号PSYNCおよびパケットデータのビットクロックBTCKに基づいて、各データを分離するためのタイミング信号LCI1SYNC、LCI2SYNC、PFXSYNC、DATASYNC、DBLAST、PERRLD、PERRSYNC(図21参照)を生成する。また、S-P変換回路420は、上記ビットクロックBTCKに同期して信号SDOをシフトし、8ビットパラレルのデータBSDATAとして、各ブロックに出力する(図29のα5)。

#### 【0154】D. 論理チャンネルの比較

S-P変換回路420の働きにより、信号LCI1SYNCがハイレベルになるタイミングでは、すでにBSDATA上に、受信されたパケットのデータLCI1が5ビットパラレルデータとして出力されている(残り3ビットは冗長データ)。チャンネル1LCI比較回路426は、信号LCI1SYNCがハイレベルのときに、BSDATA上の値と、前述のC1L1上の値とを比較し、その結果をD-FF428に出力する。双方の値が一致している場合にはハイレベルが、一致していない場合にはローレベルが、D-FF428に出力され、上記LCI1SYNCがハイレベルでかつビットクロックBTCKの立ち上がりのタイミングで、D-FF428にその比較結果が書き込まれる。

【0155】一方、信号LCI2SYNCがハイレベルになるタイミングでは、すでにBSDATA上に受信されたパケットのデータLCI2が6ビットパラレルデータとして出力されている(残り2ビットは冗長データ)。チャンネル1LCI比較回路426は、信号LCI1SYNCがローレベルのときに、BSDATA上の値と、前述のC1L2上の値とを比較し、その結果をD-FF430に出力する。双方の値が一致している場合にはハイレベルが、一致していない場合にはローレベルが、D-FF430に出力され、上記信号LCI2SY

NCがハイレベルでかつビットクロックBTCKの立ち上がりのタイミングで、D-FF430にその出力結果が書き込まれる。

【0156】ここで、受信データの論理チャンネルLCI1、LCI2の値が共に設定レジスタ424の設定値と一致した場合には、D-FF428、430からANDゲート432に共にハイレベルの信号が与えられる。また、信号HERRがハイレベルの場合には、プリデコーダ404においてBCH(16, 5)誤り訂正が不可能であったことを示し、LCI1、LCI2の受信データが信用できないことを意味する。すなわち、信号HERRがローレベルでかつLCI1、LCI2の値が共に設定値と一致したときのみ、ANDゲート432は信号CH1ENAをハイレベルにし、パケットの受信(正確にはバッファリング)を許可する。これ以外の場合は、信号CH1ENAがローレベルとなり、パケットの受信が許可されない(図29のα6)。上記と同様の動作は、チャンネル2においても行われ、この場合には、信号線C2L1、C2L2、チャンネル2LCI比較回路434、D-FF436、438、ANDゲート440が使用される。

【0157】E. プリフィックスデータの生成  
次に、チャンネルデテクタ406内のD-FF442は、信号PFXSYNCに基づいて、8ビットのプリフィックスデータ(SCC, CI, DTC)を記憶し、8ビットパラレルの信号PRFXDTとして出力する(図29のα7)。また、D-FF444は、信号PERRLDに基づいて、SDSC(272, 190)誤り訂正不可能を示す信号PERRを記憶し、信号PERRLTとして出力する。信号PERRは、上記パケットの最後のデータとして、プリデコーダ404から与えられる。

【0158】F. データのバッファリング  
チャンネルデコーダ408には、全く同じ回路が2回路用意されており、各々チャンネルデテクタ408からの信号CH1ENAおよびCH2ENAに基づいて動作する。その内の1つであるチャンネル1デコーダ/バッファ408a内のチャンネル1データコントローラ460(図14参照)は、以下の順序でデータのバッファリングを制御する。

- 40 a) 先頭パケットの検出…初期化バッファリング開始
  - b) 定常動作  
データメモリ454への書き込みおよびアドレス制御  
CRC演算、データサイズカウンタ462のデクリメント  
CIカウントによるパケットの連続性チェック  
プリフィックスメモリ452への書き込みおよびアドレス制御
  - c) 終端パケットの検出…バッファリング停止
- 以下、上記各動作を順番に説明する。

50 【0159】a) 先頭パケットの検出

データコントローラ450内のデータグループシーケンサ460（図15参照）は、図30の階層4に示すデータグループを構成すべく、PRFXDT上のプリフィックスデータのDTCに基づいて、先頭の packets と終端の packets を検出する。例えば、前述の信号CH1ENAがハイレベルであり、packetsの受信が許可されていても、まだ先頭packetsが検出されていない状態では（先頭packets未検出状態）、packetsのバッファリングは行わない。先頭packetsを検出すると、データグループシーケンサ460は、信号CILDLをCIカウンタ464に与え、信号CRCCLRをCRC回路468に与えて、これら回路を初期化する。また、データグループの先頭から3～5バイト目に受信されるデータグループサイズをデータサイズカウンタ462内にストアすべく、信号DSLDDを出力する。

#### 【0160】b) 定常動作

データメモリ454への書き込みおよびアドレス制御  
データグループシーケンサ460は、信号DATASYNCに基づいて信号DFIFOWRを生成し、データFIFOインターフェース474に与える。データFIFOインターフェース474内のタイミング発生回路520（図18参照）は、信号DFIFOWRに基づいて信号DWRENを生成する。信号DWRENの生成には、CPU30からのデータリードと衝突しないタイミングが選ばれる。CPU30から出力された信号SYSCKがハイレベルの期間はCPU30がデータリードを行う期間であり、ローレベルの期間はメモリをアクセスしない期間であるため、この信号SYSCKに基づいてタイミング発生回路520が信号DWRENを生成する。信号DWRENは、インバータ528を介して、チャンネル1データメモリ454（図14参照）にデータの書き込み信号C1DWEとして与えられる。

【0161】データラッチ522は、ビットクロックBTCKに基づいてデータBSDATAを常にラッチしており、前述のタイミング調整によって書き込みタイミングに遅延が生じてデータが消失しないようにこれを保持する。ライト用ループカウンタ524は、書き込み時のアドレスを記憶している。アドレスマルチプレクサ532は、書き込みアドレスを、信号C1DADRとしてデータメモリ454に与える。

【0162】これによって、データメモリ454にアドレス、データ、書き込み信号が与えられるため、当該データメモリ454に受信データが順次記憶される。また、リード用ループカウンタ524は、信号DWRENに基づいてアップカウントされ、データメモリ454への書き込みアドレスを歩進する。信号DWRENは、インバータ540を介して信号RDLATG1となり、前述のタイミング調整時のCPUリードデータ保持用ラッチのための書き込み禁止信号として使用される。これら書き込みタイミングの詳細は、図22に示されている。

【0163】CRC演算、データサイズカウンタ462のデクリメント

図15のデータグループシーケンサ460は、信号DATASYNCに基づいて信号CRCNXTを生成し、CRC回路468に与える。CRC回路468は、信号CRCNXTを受けて、BSDATA上のデータグループの各データを順次CRC演算する。

【0164】また、データグループシーケンサ460は、信号CRCNXTに同期して、信号DSDECをデータサイズカウンタ462に出力する。データサイズカウンタ462は、この信号DSDECを受けて、内部のデータサイズ値をデクリメントする。データサイズ値が0になると、データサイズカウンタ462は信号DSEQを出力し、データグループシーケンサ460に与える。データグループシーケンサ460は、信号DSEQを受けると、データグループの全データを受信し終えたと判断し、信号CRCOKLDをCRC回路468に与える。これによって、CRC回路468は、CRCの演算結果を内部レジスタにストアし、その結果をORゲート470に出力する。

【0165】ORゲート470は、前述の信号PERRLTと、CRC演算結果とをORすることにより、どちらかにエラーがあった場合に、信号CRCERRをハイレベルにしてプリフィックスFIFOインターフェース472に与える。この信号CRCERRは、受信されたプリフィックスデータSCC、DTCと共に、図14のチャンネル1プリフィックスメモリ452に与えられて記憶されるが、CRCの演算結果については、データグループの終了時（すなわち終端packets）においてのみ有効なフラグとなる。

【0166】CIカウントによるpacketsの連続性チェック

データグループシーケンサ460は、信号PERRSYNCに基づいて、信号CICNTENを生成する。CIカウンタ464は、先頭packetsで初期化したCI値を、packets毎にインクリメントし、その結果をEXOR（排他的論理和）ゲート466に出力する。EXORゲート466は、CIカウンタ464からのカウント値と、プリフィックスとして受信されるCI値とが一致するかどうかを検出し、一致しない場合には信号CIERRをハイレベルにしてプリフィックスFIFOインターフェース472に出力する。一致しない場合は、受信状態が不良で、エラー訂正不可能なpacketsが発生し、データグループにpackets抜けが発生したことを意味する。この信号CIERRは、信号CRCERRと共に、受信されたプリフィックスデータSCC、DTCと統合され、6ビットパラレルの信号C1PDとなって、図14のチャンネル1プリフィックスメモリ452に与えられて記憶される。

【0167】プリフィックスメモリ452への書き込み

およびアドレス制御

図15のデータグループシーケンサ460は、信号PERRSYNCに基づいて信号PFI FOWRを生成し、プリフィックスFIFOインターフェース472に与える。プリフィックスFIFOインターフェース472内のタイミング発生回路500(図17参照)は、信号PFI FOWRに基づいて信号PWRENを生成する。信号PWRENの生成には、CPU30からのデータリードと衝突しないタイミングが選ばれる。そこで、前述の信号DWRENの場合と同様に、信号SYSCCKに基づいてタイミング発生回路500が信号PWRENを生成する。信号PWRENは、インバータ508を介して信号C1PWEとなり、図14のチャンネル1プリフィックスメモリ452に、プリフィックスデータの書き込み信号として与えられる。

【0168】図17のプリフィックスデータラッチ502は、ピットクロックBTCKに基づいて信号PRFXDTを常にラッチしており、前述のタイミング調整によって書き込みタイミングに遅延が生じてもプリフィックスデータが消失しないようにこれを保持する。ライト用ループカウンタ504は、書き込み時のアドレスを記憶している。アドレスマルチプレクサ512は、書き込みアドレスを信号C1PADRとしてプリフィックスメモリ452に与える。これによって、プリフィックスメモリ452にアドレス、データ、書き込み信号が与えられるため、プリフィックスメモリ452に順次受信プリフィックスデータが記憶される。

【0169】また、ライト用ループカウンタ504は、信号C1PWEに基づいてインクリメントされ、図14のチャンネル1プリフィックスメモリ452への書き込みアドレスを歩進する。信号PWRENは、インバータ518を介して信号PFLATG1となり、前述のタイミング調整時のCPUリードデータ保持用ラッチのための書き込み禁止信号として使用される。これら書き込みタイミングの詳細は、図22に示される。また、プリフィックスデータ数カウンタ514は、信号PWRENに基づいてインクリメントされ、現在メモリに蓄えているパケット数をPFCNT1として出力する。

【0170】c) 終端パケットの検出

図15のデータグループシーケンサ460は、プリフィックスのDTCに基づいて終端パケットを検出する。データグループがスタイル1の場合には、前述のようにデータグループサイズのカウンタによって信号CRCOKLDを生成するが、データグループがスタイル2の場合には、この終端パケットを示すDTCと信号DBLASTとに基づいて、信号CRCOKLDを生成する。パケット抜けやCRCエラーの場合を考慮し、終端パケットが検出されたとしても、CPU30から受信許可フラグ460cを禁止状態にセットしに行かない限り、本システムは、再度同じデータを受信し続ける。

【0171】G. 各データのリード

CPU30が特定のI/Oアドレスをリードすると、それに対応して、ホストインターフェース410内のアドレスデコーダ410cがデコード信号を生成する。例えば、図14のチャンネル1プリフィックスメモリ452からデータをリードする場合には、デコード信号PFI FORDがチャンネルデコーダ408に出力される。

【0172】チャンネルデコーダ408内のデータセレクト456は、信号PFI FORDを受けて、チャンネル1プリフィックスメモリ452からのプリフィックスデータC1PODをD-F F480で一旦ラッチした信号(すなわち、バッファリングされたプリフィックスデータ)をデータバスB上に出力する(図29のα8)。これによって、CPU30は、プリフィックスデータをリードできる。

【0173】また、信号PFI FORDは、プリフィックスFIFOインターフェース472にも与えられ、タイミング発生回路500が、これに基づいて信号PRDENおよびPDWを生成する。信号PRDENは、リード用ループカウンタ506に作用し、リード用アドレスをインクリメントさせる。また、信号PDWは、プリフィックスデータ数カウンタ514に作用し、バッファリングしているパケットのカウンタ値をダウンカウンタさせる。アドレスマルチプレクサ512は、データ書き込み時以外、常にリード用ループカウンタ506からのリードアドレス値を信号C1PADRとして出力する。

【0174】同様に、CPU30は、チャンネル1データメモリ454の内容、プリフィックスデータ数カウンタ514の内容、ORループ483の内容をリードできる(図29のα9~α11)。

【0175】上記のようにして、CPU30は、バッファメモリ452、454から受信したデータを抜き出し、例えば本体内のワークRAM32にTCDを展開する。次に、CPU30は、プリフィックスデータのエラーフラグによって、展開したTCDに受信エラーが無いことを確認の後、これをメニューデータとしてTVモニタ22の画面上に表示する。

【0176】(5) 番組の選択動作

オペレータは、TVモニタ上に表示されたメニューを見て、受信したい番組名を選択する。CPU30は、オペレータによって選択された番組名を、対応する論理チャンネルデータに変換し、これを内部メモリに一旦記憶する。また、CPU30は、ワークRAM32上に展開されているTCDから、当該選択された番組に対応する使用メモリ容量、実行メモリマップ構成を抽出し、これに基づいてメモリマップ上での各メモリの配置を決定する。

【0177】(6) メモリマップ上での各メモリの配置動作

次に、CPU30は、図21に示すメモリマップ上に、

疑似SRAM74と、フラッシュメモリ76および106とを配置する。ここでは、一例として、疑似SRAM74をメモリマップ上に配置する動作の詳細について説明する。

【0178】まず、CPU30は、疑似SRAM74のマッピング設定のためのI/OポートアドレスをアドレスバスAに出力するとともに、データバスAに設定値を出力し、制御信号Aに含まれる信号CPUWRBをローレベルにセットする。例えば、図35に示す位置に疑似SRAM74を配置する場合、RAMON=1、RAMMAP0=0、RAMMAP1=1、RAMMAP2=0に設定しなければならない。従って、CPU30は、アドレスバスA上にRAMONを示すI/Oポートアドレス値を出力し、データバス上にデータ「80H」を出力する。

【0179】マッピングコントローラ70内のアドレスデコーダ112（図7参照）は、上記データバスA上のRAMONのI/Oポートアドレス値をデコードすることにより、デコード信号FE3をハイレベルにし、コントロールレジスタ110に出力する。データバスのD7は、バッファ118を介して信号CD7Iとしてコントロールレジスタ110に与えられる。このとき、データの値が「80H」であるため、CD7Iはハイレベルの信号となる。また、上記信号CPUWRBもコントロールレジスタ110に与えられる。

【0180】前述したように、コントロールレジスタ110の内部には、前後段2個のD-FFとANDゲートとから成る複数のレジスタブロック190~300が存在する（図11参照）。また、リセット信号は、各レジスタブロックに要求される初期値に対応して、各レジスタブロックのD-FFのセット端子またはクリア端子に、個別的に与えられる。信号FE3は、レジスタブロック200内に存在する前段のD-FF202の端子CEに書き込み許可信号として与えられる。また、信号CPUWRBは、同D-FF202のクロック端子に書き込み信号として与えられる。これによって、D-FF202には、CD7Iの値（ハイレベルすなわち論理1）が書き込まれる。

【0181】次に、CPU30は、アドレスバスA上にRAMMAP0を示すI/Oポートアドレス値を出力し、データバス上にデータ「00H」を出力する。上記と同様にして、RAMMAP0に対応するレジスタブロック210内のD-FF212は、信号FE4を受けて、ローレベル（論理0）を記憶する。

【0182】次に、CPU30は、アドレスバスA上にRAMMAP1を示すI/Oポートアドレス値を出力し、データバス上にデータ「80H」を出力する。上記と同様にして、RAMMAP1に対応するレジスタブロック220内のD-FF222は、信号FE5を受けて、ハイレベル（論理1）が書き込まれる。

【0183】次に、CPU30は、アドレスバスA上にRAMMAP2を示すI/Oポートアドレス値を出力し、データバス上にデータ「00H」を出力する。上記と同様にして、RAMMAP2に対応するレジスタブロック230内のD-FF232は、信号FE6を受けて、ローレベル（論理0）が書き込まれる。

【0184】さらに、CPU30は、アドレスバスA上にENMAPSTを示すI/Oポートアドレス値を出力し、データバス上にデータ「00H」を出力する。このデータはダミーデータであり、特に意味は無い。ENMAPSTへ書き込みを行うという動作によって、信号FE14がアドレスデコーダ112から各レジスタブロックの後段のD-FF194, 204, 214, ..., 304のCE端子に与えられる。同時に、信号CPUWRBが各D-FF194, 204, 214, ..., 304クロック入力端子に与えられる。これによって、D-FF192, 202, 212, ..., 302に書き込まれたデータが、後段のD-FF194, 204, 214, ..., 304にそっくり移される。この時点で、先ほど書き込んだRAMON、RAMMAP0、RAMMAP1、RAMMAP2のデータが、信号RAMONおよび信号RAMMAP0~2として、マップデコーダ120に出力される。このように、レジスタブロックに順次データを設定した後、一度に全設定値を有効化することにより、マッピング変更時の遷移期間中のメモリ配置を気にする必要がなくなる。その結果、容易にマッピング変更が行えるメモリマッピング変更システムを実現できる。

【0185】上記のようにして各レジスタブロック190~300に設定された内容は、マップデコーダ120に inputs され、疑似SRAM74のためのチップイネーブル信号CE1Bの生成に使用される。

【0186】なお、他のメモリ、すなわちフラッシュメモリ76および106をメモリマップ上に配置する動作も上記と同様にして行われる。参考のために、図32~図34に各メモリ配置のために必要となる条件を示しておく。すなわち、図32は、PS-RAMデコーダ162において、疑似SRAM74のチップイネーブル信号CE1Bがアクティブ（ローレベル）になる条件を示している。また、図33は、フラッシュデコーダ164において、フラッシュメモリA76のチップイネーブル信号CE3Bがアクティブ（ローレベル）になる条件を示している。また、図34は、メモリユニットデコーダ166において、フラッシュメモリB106のチップイネーブル信号CE4Bがアクティブ（ローレベル）になる条件を示している。

【0187】（7）実データの受信動作  
次に、CPU30は、実際データ（TCD以外のデータ）を受信する。実データの受信動作は、基本的には前述のメニューデータの受信動作と同様である。すなわち、ユーザによって選択された番組に対応する論理チャ



ンネルが、CPU30からデータ放送デコード82に対してセットされる。データ放送デコード82は、受信した放送データの中から、セットされた論理チャンネルに対応するパケットのみを抜き出し、バッファリングする。CPU30は、バッファメモリ452、454に格納されたデータを、メモリマップ上に配置されたメモリ（例えば、疑似SRAM74にDMA転送する。

#### (8) エラー検出動作

次に、メニューデータまたは実データを受信する際に実行されるエラー検出動作について説明する。本実施例では、誤りのない受信データを得るために、BCH誤り訂正と、SDSC誤り訂正と、CIカウンタ464によるパケットの連続性チェックと、CRC回路468によるCRC演算とを行っている。BCH誤り訂正は、受信データの各パケットのヘッダ部に含まれている誤りを訂正するために行われる。SDSC誤り訂正は、受信パケットデータの本体（ペイロード部）に含まれている誤りを訂正するために行われる。

【0188】BCH誤り訂正において、誤り訂正をしきれずにエラーが残った場合、図12のBCH(16、5)誤り訂正回路404dは、信号HERRを出力する。この信号HERRは、チャンネルデテクタ406内のANDゲート432、440（図13参照）を閉じるように作用する。その結果、信号CH1ENA、CH2ENAをローレベルに強制する。これによって、図15のデータグループシーケンサ460が不能動作され、許可フラグレジスタ460bがオンされない。従って、図14のバッファメモリ452、454へのデータの書き込みが許可されない。

【0189】SDSC誤り訂正において、誤り訂正をしきれずにエラーが残った場合、図12のSDSC(272、190)誤り訂正回路404eは、信号PERRを出力する。この信号PERRは、各パケットの終端部において図13のD-FF444にラッチされた後、信号PERRLTとしてデータコントローラ450におけるORゲート470で、CRC演算結果（CRCの演算結果については、データグループの終了時、すなわち終端パケットにおいてのみ有効なフラグとなる）と統合されて、信号CRCERRとなる。

【0190】パケット抜けによって、受信データのCI値とCIカウンタ464のカウント値との不一致が生じた場合、EXORゲート466は、信号CIERRをハイレベルにする。

【0191】上記信号CRCERRおよびCIERRは、プリフィックスFIFOインターフェース472において、プリフィックスデータ内に挿入され、6ビットのプリフィックスデータC1PIDとなってチャンネル1プリフィックスメモリ452に格納される。CPU30が、このチャンネル1プリフィックスメモリ452からデータを読み出すとき、チャンネル1データセクタ

456内のORループ483が各ビット毎に時間軸方向のORをとり、その結果をD-FFに記憶する。すなわち、各パケットのプリフィックスデータを時間軸方向に並べた複数のビット列において、信号CRCERRに対応するビット列中に1つでも論理1があると、ORループ483内のD-FF486の対応する記憶部が論理1を保持する。また、各パケットのプリフィックスデータを時間軸方向に並べた複数のビット列において、信号CIERRに対応するビット列中に1つでも論理1があると、D-FF486の対応する記憶部が論理1を保持する。従って、CPU30は、プリフィックスデータ読み出し終了後、D-FF486の記憶情報を読み出すことにより、バッファリングされている受信パケットに、SDSCエラー、CRCエラー、不連続性エラーが生じているか否かを知ることができる。このような手法によれば、各パケット毎にプリフィックスデータを調べる場合に比べて、プログラムの負担を大幅に軽減できる。

【0192】なお、上記説明は、主としてチャンネル1に対する受信動作について述べたが、チャンネル2についても上記と同様の動作で受信が可能なことは言うまでもない。このように2つのチャンネル回路を並列的に設けることにより、一方のチャンネルでプログラムデータ等を受信しつつ、他方のチャンネルで音声データや表示データを受信することが可能となり、それをユーザに流すことで、受信中の退屈感を紛らわせることができる。

【0193】図35は、データ放送を受信する場合に好適なメモリ配置を示している。メモリマップの制御領域以外の部分には、フラッシュメモリB106が全面的に配置されている。ただし、図示していないが、バンク7E、7Fには特別領域が予め設定されているため、フラッシュメモリB106は配置されていない。そのフラッシュメモリB106上にマスクROM72がROMON0=1によって配置され、疑似SRAM74がRAMMAP0=0、RAMMAP1=1、RAMMAP2=0、RAMON=1によって配置される。また、フラッシュメモリA76は、FMON=1、FMMAPO=0、FMMAPI=1によって図の位置に配置される。

【0194】本実施例で使用したCPU30の特性として、割り込みベクトルがバンク0のFFF0H以降にセットされるため、マスクROM72を図35の位置に置くことにより、マスクROM72内の割り込みルーチンが使用可能となり、データ放送の受信が適切に行われる。CPU30は、受信データを、疑似SRAM74／フラッシュメモリA76／フラッシュメモリB106の任意のアドレスに書き込むことができる。

【0195】ここで、疑似SRAM74の下に隠れているフラッシュメモリB106の一部分は、RAMON=0として、疑似SRAM74をメモリマップ上から消失させるか、またはRAMMAP0~2の値を変更して、疑似SRAM74の位置を移動することによりアクセス

10

20

30

40

50

できるようになる。さらに、RAMMAP0=0、RAMMAP1=0、RAMMAP2=1と設定することにより、疑似SRAM74を、フラッシュメモリA76の80~8Fバンクにセットすることもできる。この場合、フラッシュメモリB106、フラッシュメモリA76、疑似SRAM74は、この領域において3層のメモリレイヤを持つことになる。

【0196】なお、上記図35のメモリ配置は、マスクROM72に文字や静止画を表示するプログラムが予め内蔵されている場合、あるいは文字および静止画の放送データを受信しながら、これを画面に表示する場合に好適なメモリ配置である。

【0197】図36は、フラッシュメモリA76内に記憶したゲームプログラムを動作させる場合に好適なメモリ配置を示す。この場合、CPU30がゲーム用の割り込みベクトルを使用できるように、フラッシュメモリA76を第0バンクに設定する必要があるため、図35に示すようなマスクROMエリアは排除し(ROMON0=0)、フラッシュメモリA76の位置を変更しなければならない(FMMAP0=0、FMMAP1=0)。しかし、ゲームプログラムからマスクROM72内に含まれる画像、音声制御サブルーチンを使用できれば便利であるため、ただ単にマスクROM72のメモリエリアを排除するだけでなく、マスクROM72を図36の位置に移動することが必要となる。

【0198】この場合には、ROMON1=1と設定すればよい。ただし、上記画像、音声制御サブルーチンは、アドレス的にリロケータブルであること(プログラム上で絶対アドレスとして記述されていないこと)が要求される。また、マスクROM72の移動によって隠れてしまったフラッシュメモリB106のバンクA0~BFの領域にアクセスしたい場合には、ROMON1=0とすることによりアクセスが可能となる。このように、各メモリをプログラムの状況に応じてメモリマップ上の最適の位置に配置することができる。

【0199】ここで最適の位置とは、単にプログラム実行の邪魔にならない位置という意味もあるが、実開平4-78619号公報に開示されるように、CPUアドレスに対応してCPUクロック周波数が変化する場合に、高速のアクセススピードに対応するアドレス配置を指定できるという意味も含んでいる。また、逆にあえて低速のアクセススピードに対応するアドレスに配置し、不要輻射を低減するようメモリ配置を選択することもできる。さらに、従来のバンク設定のように、特定のバンク幅に影響されず、広範囲に渡って全メモリエリア(例えば、マスクROMエリアの16Mビット全て)がメモリマップ上に出現するため、データ転送時に頻繁にバンク切り替えを行う必要が無いという利点もある。

【0200】図35のメモリ配置でデータを受信し、受信後図36のメモリ配置でデータを表示(または、デー

タをプログラムとして実行)する場合のフローチャートを図37に示す。以下、この図37について説明する。

【0201】通常、リセット時および電源投入時には、マスクROM72は、図35に示す位置に再配置されている(ステップS101)。次に、CPU30は、TVモニタ22上に初期画面(メーカー名や本システムのタイトル等)を表示した後(ステップS102)、マスクROM72に格納されたメニュー受信ルーチンを読み出して起動する(ステップS103)。このメニュー受信ルーチンにおいて、データ放送からのメニューデータの受信が行われるが、受信されたメニューデータは、通常、メモリマップ上の制御領域にあるワークRAM32にストアされる(ステップS104)。なお、メニューデータを疑似SRAM74に記憶させたい場合には、疑似SRAM74をメモリマップ上に配置してやる必要がある。

【0202】メニューデータは、受信可能なデータ名と、それに対応する使用メモリ容量および実行メモリマップ構成とを含む。データ名がTVモニタ22の画面上に表示され、ユーザーによってその中の1つが選択される(ステップS105)。ユーザーによって特定のデータが選択されると、まず、それを受信するためのメモリ配置がCPU30によって決定され(ステップS106)、当該決定に従って疑似SRAM74およびフラッシュメモリA76、B106がメモリマップ上に配置される(ステップS107)。このとき、フラッシュメモリB106は、メモリマップ上全面に渡ってバックグラウンドとして配置される。フラッシュメモリB106の記憶容量がメモリマップ容量よりも小さい場合、フラッシュメモリB106の同一のメモリエイメージがメモリマップ上に複数個出現する。このため、他のメモリがフラッシュメモリB106の上に重なって配置されても、他のメモリの隙間からフラッシュメモリB106の任意のアドレスをアクセスできる可能性が高くなる。

【0203】次に、実際のデータが受信されるが(ステップS108)、フラッシュメモリへの書き込み速度が遅く、DMA転送に対応できない場合には、まず疑似SRAM74にデータをDMA転送してから、順次フラッシュメモリへ書き込むという作業が必要になる(ステップS109)。しかし、疑似SRAM74とフラッシュメモリの書き込まれるべきエリアとが重なっている場合には、書き込みができないため、疑似SRAM74を当該エリアに重ならない位置に移動させる必要がある(ステップS110~S112)。フラッシュメモリには、カートリッジ6に内蔵されるフラッシュメモリA76と、メモリユニット8に内蔵されるフラッシュメモリB106とのどちらを用いてもよい。疑似SRAM74が問題の無い位置にあることを確認した後、データはフラッシュメモリに書き込まれる(ステップS113)。以上の動作が実際のデータを全て受信し終わるまで繰り返

し実行される（ステップS108～S114）。

【0204】次に、受信した実際のデータをプログラムとして実行するに際して、疑似SRAM74の現状の配置が問題になるかどうかチェックされる（ステップS115）。これは、マスクROM72上で現在動作しているプログラムをフラッシュメモリ上で動作させるように切り替える場合に、動作中のメモリの配置を切り替えると暴走の恐れがあるためである。問題がある場合、一旦、プログラムを疑似SRAM74上で動作させ、マップを切り替えてからフラッシュメモリに制御を移すという手順が採用される。ただし、フラッシュメモリが疑似SRAM74の下に隠れている場合（ステップS116）は、フラッシュメモリへの移行が不可能であるため、予め不具合の生じない位置に疑似SRAM74を移動しておく（ステップS117）。疑似SRAM74の位置が決定されると、準備プログラムが転送、実行され、マップの切り替え、フラッシュメモリへのジャンプが行われる（ステップS118～S120）。

【0205】その後は、データ放送から受信されたプログラムおよびデータに従ってCPU30は動作し、必要に応じてメモリマップの配置を切り替える（ステップS121）。

【0206】ここで、本実施例では、メモリマップの配置決定方法として、3種類の方法が準備されていることがわかる。すなわち、第1番目の方法は、マスクROM72内に記憶されているメモリマップ配置データにより決定される方法である。第2番目の方法は、データ放送のメニューデータに含まれ、データ受信時に必要なメモリマップ配置を示すデータによって決定される方法である。第3番目の方法は、受信された実際のデータおよびプログラムに含まれ、これを表示あるいは実行する際に必要となるメモリマップ配置を示す実行データにより決定される方法である。特に、第2番目および第3番目の方法は、放送されるメニューデータおよび実行データに基づいてメモリマップの配置を設定できるため、放送されるデータの容量、種類によって最適な配置を放送時に指定できる。

【0207】図38は、図37におけるサブルーチンステップS103またはS108において、CPU30によって実行される動作を示したフローチャートである。以下、この図38を参照して、データ放送受信時におけるCPU30の動作を説明する。

【0208】まず、CPU30は、アドレスバスBにアドレス信号を出力し、データバスに論理チャンネルデータを出力し、制御信号Bに含まれる信号PAWRをローレベルにセットすることにより、図5のデータ放送デコーダ80に論理チャンネルをセットする（ステップS201）。次に、CPU30は、受信プログラム（マスクROM72に格納されている）に従って、次フレームでTVモニタ22に出力すべき画像・音声（例えば、ユー

ザの待機を促すメッセージ）のデータの処理を、ビデオプロセッサ40およびオーディオプロセッサ46と協働して行う（ステップS202）。次に、CPU30は、ビデオプロセッサ40からブランキング期間における割り込み指示NMIが来るのを待機する（ステップS203）。当該割り込み指示NMIを受け取ると、CPU30は、図17のプリフィックスデータ数カウンタ514をアクセスし、そこに格納されているプリフィックスデータ数を確認する（ステップS204）。CPU30は、プリフィックスデータ数の確認結果に基づいて、データメモリ454内にデータが格納されているか否かを判断する（ステップS205）。データメモリ454内にデータが格納されていない場合は、転送すべきデータが溜まっていないと判断し、ステップS202の動作に戻る。

【0209】一方、データメモリ454内にデータが格納されている場合、CPU30は、図16のORループ483に読み出し指示を与えることにより、D-FF486をリセットする（ステップS206）。次に、CPU30は、プリフィックスメモリ452内にバッファリングされたプリフィックスデータをDMA転送する（ステップS207）。プリフィックスデータの転送先は、前述のステップS103では、ワークRAM32であり、ステップS108では、疑似SRAM74である。次に、CPU30は、ORループ483内のD-FF486に記憶されたエラーフラグの状態を確認する（ステップS208）。CPU30は、このエラーフラグの状態に基づいて、何らかのエラーが生じているか否かを判断する（ステップS209）。前述したように、D-FF486に記憶されたエラーフラグが論理1の場合、データメモリ454内に格納されているデータパケットのいずれかにエラーが生じている。

【0210】当該エラーフラグが論理1で、何らかのエラーが発生している場合、CPU30は、すでにワークRAM32または疑似SRAM74に転送されたプリフィックスデータを参照することにより、どのパケットにどのようなエラーが生じているかを確認し、再受信処理の準備を行う（ステップS210）。なお、ここで確認されるエラーは、SDSCエラーと、CIエラーである。ステップS209でエラーが検出されない場合（D-FF486に記憶されたエラーフラグが論理0の場合）、またはステップS210の処理の後、CPU30は、データメモリ454内にバッファリングされたデータグループデータをDMA転送する（ステップS211）。データグループデータの転送先は、前述のステップS103では、ワークRAM32であり、ステップS108では、疑似SRAM74である。次に、CPU30は、データグループデータの全てを転送したか否かを判断する（ステップS212）。この判断は、終端パケットの検出に基づいて行われる。もし、終端パケットの

転送が終了していない場合、CPU30は、再びステップS202の動作に戻り、残りのパケットデータの受信およびDMA転送を繰り返す。

【0211】一方、終端パケットの転送が終了すると、CPU30は、ワークRAM32または疑似SRAM74に記憶されたプリフィックスデータ中のCRCフラグ（SDSCエラーを示すフラグの内、終端パケットに対応するフラグ）をチェックし（ステップS213）、受信データ内にCRCエラーが生じているか否かを判断する（ステップS214）。CRCエラーが生じている場合、CPU30は、再受信処理の準備を行う（ステップS215）。CRCエラーが生じていない場合、またはステップS215の再受信準備が終了した後、CPU30は、データ放送受信処理を終了する。

【0212】ところで、上記ステップS209において、受信データにエラーのあることが発見された場合の対処方法としては、エラーのあるデータパケットのみを再受信して転送する第1の方法と、データグループデータ全体を再受信して転送する第2の方法とがある。第1の方法は、短時間で処理が終了する反面、データ放送デコーダ82内でCRC演算が行えないため、データ転送後にCPU30が転送先のメモリの記憶内容からCRC演算を行う必要がある。その結果、CRCエラーが検出されると、再びデータグループデータ全体を再受信して転送する必要がある。そのため、本システムの受信性能や、現在の受信状態を考慮して、いずれの方法を選択するかを決めることが好ましい。例えば、雷によるインパルス性ノイズの混入時には、第1の方法が好ましい。また、降雨によって受信データ全体のS/N比が低下している場合は、第2の方法が好ましい。なお、上記ステップS214において、CRCエラーが発見された場合は、データグループデータ全体の再受信および転送が必要になるため、第2の方法により対処される。

【0213】図39は、図7に示すマップデコーダ120の他の構成例を示すブロック図である。なお、この図39は、優先順位レジスタを設けることにより、各メモリ回路の優先順位をプログラマブルに設定することが可能なマップデコーダを例示している。

【0214】図39において、本マップデコーダは、図10に示すマップデコーダの構成と同様に、プリデコード信号CE2B'を出力するROMデコーダ160'と、プリデコード信号CE1B'を出力するPS-RAMデコーダ162'と、プリデコード信号CE3B'を出力するフラッシュデコーダ164'と、プリデコード信号CE4B'を出力するメモリユニットデコーダ166'とを備えている。しかしながら、これらデコーダ160'、162'、164'、166'は、前段のデコーダの出力が、後段のデコーダにデコード条件として与えられていない。従って、プリデコード信号CE1B'、CE2B'、CE3B'、CE4B'は、優先順

位を持っておらず、メモリマップ上の同じ位置に複数のメモリが配置されている場合、CPU30が当該位置をアクセスすると、複数のプリデコード信号が同時にアクティブになる可能性がある。そこで、優先順位を持たないプリデコード信号に対して、優先順位を与えるために、アドレスデコーダ170と、優先順位設定レジスタ172と、セクタA174と、セクタB176と、優先順位回路178とがさらに設けられる。

【0215】次に、図39に示すマップデコーダの動作を説明する。CPU30は、優先順位設定レジスタ172に優先順位データ（マスクROM72/疑似SRAM74/フラッシュメモリA76/フラッシュメモリB106の優先順位を切り替えるためのデータ）を書き込む際に、アドレスバスA上に優先順位設定レジスタ172のI/Oポートアドレスを示すアドレス信号を出力し、データバス上に当該優先順位データを出力する。さらに、CPU30は、優先順位設定レジスタ172に、書き込み信号CPUWRBを出力する。

【0216】アドレスデコーダ170は、アドレスバスA上のアドレス信号をデコードして書き込み許可信号PRCEを出力し、優先順位設定レジスタ172に与える。優先順位設定レジスタ172は、アドレスデコーダ170からの書き込み許可信号PRCEおよびCPU30からの書き込み信号CPUWRBにตอบสนองして、データバス上の優先順位設定データを取り込んで記憶する。

【0217】セクタA174は、「4→1」セクタを4回路内蔵しており、優先順位設定レジスタ172からの優先順位データに基づいて、各デコーダからのプリデコード信号の配列を切り替え、その優先順位を変更する。すなわち、セクタA174は、優先順位の高いものが上から順番に並ぶように、プリデコード信号の配列を切り替える。

【0218】優先順位回路178は、セクタA174を上から順番（すなわち、優先順位の高い方から順番に）にサーチし、最初にアクティブになっている信号を検出し、当該信号をアクティブ（ローレベル）、他の信号を非アクティブ（ハイレベル）の状態にしてセクタB176に出力する。また、セクタA174から入力する信号がすべて非アクティブであった場合には、セクタB176へ出力する信号を全て非アクティブにする。

【0219】セクタB176は、セクタAと同様の構成を有し、セクタA174によって優先順位変更のために組み替えられた配線状態を元に戻し、各メモリ回路のためのチップイネーブル信号CE1B、CE2B、CE3B、CE4Bを出力する。優先順位回路178によって4本の信号線の内の1本だけがアクティブになる（または全てが非アクティブになる）ため、セクタB176から出力される各メモリ回路のチップイネーブル信号は、同時に2本以上がアクティブになることは無

い。また、優先順位設定レジスタ172に書き込む優先順位データを変更することにより、容易に各メモリ回路のアクセス優先順位を変えることができる。

#### 【0220】

【発明の効果】請求項1の発明によれば、パケット単位でエラーをチェックすることなく、転送データにエラーに存在するか否かを容易に確認することができるので、エラーチェックに要するCPUの負担を大幅に軽減することができる。

【0221】請求項2の発明によれば、バッファリング手段に記憶されたエラーフラグが、第1および第2のデータと共に、情報処理装置で使用されるメモリ回路に転送して格納されるので、CPUは、エラー確認手段がエラーの存在を確認したとき、メモリ回路に格納されたエラーフラグを参照することで、どのパケットにエラーが生じているかを容易に知ることができる。

【0222】請求項3の発明によれば、第1のデータに含まれている区切り用識別子を、エラーフラグと共に、時間軸方向に論理和演算し、この論理和演算結果に基づいて、バッファリング手段に記憶された第2のデータに、先頭パケットまたは終端パケットが存在するか否かを確認するようにしているので、CPUは、パケット単位で区切り用識別子をチェックすることなく、データの先頭と終端とを知ることができる。

【0223】請求項4の発明によれば、天候等により様々なエラーが発生し易い放送衛星からの受信データを処理するようにしているので、エラーチェックによるCPUの負担軽減効果が特に顕著に現れる。

#### 【図面の簡単な説明】

【図1】本発明の一実施例に係る衛星データ放送受信装置の構成を示す外観斜視図である。

【図2】図1の衛星データ放送受信装置を用いて構成された衛星データ放送受信システムの全体構成を示す外観斜視図である。

【図3】図2におけるゲーム機本体2の内部構成を示すブロック図である。

【図4】図2における衛星データ放送受信カートリッジ6の内部構成を示すブロック図である。

【図5】図2における衛星データ放送受信アダプタ4の内部構成を示すブロック図である。

【図6】図2におけるメモリユニット8の内部構成を示すブロック図である。

【図7】図4におけるマッピングコントローラ70の内部構成を示すブロック図である。

【図8】図7におけるIRQコントローラ126の内部構成を示すブロック図である。

【図9】図7におけるPSRAMコントローラ124の内部構成を示すブロック図である。

【図10】図7におけるマップデコーダ120の内部構成を示すブロック図である。

【図11】図7におけるコントロールレジスタ110の内部構成を示すブロック図である。

【図12】図5におけるデータ放送デコーダ82の内部構成を示すブロック図である。

【図13】図12におけるチャンネルデテクタ406の内部構成の詳細を示すブロック図である。

【図14】図12におけるチャンネル1用のデコーダ/バッファ408aの内部構成を示している。

10 【図15】図14におけるデータコントローラ450の内部構成の詳細を示すブロック図である。

【図16】図14におけるデータセクタ456の内部構成の詳細を示すブロック図である。

【図17】図15におけるプリフィックスFIFOインターフェース472の内部構成の詳細を示すブロック図である。

【図18】図15におけるデータFIFOインターフェース474の内部構成の詳細を示すブロック図である。

【図19】図18におけるEMP/FULデテクタ538の内部構成の詳細を示すブロック図である。

20 【図20】図19に示すEMP/FULデテクタ538の動作を、各入力条件別に示す図である。

【図21】図13のビットカウンタ422から出力される各種信号のタイミングを示すタイムチャートである。

【図22】図14のデータメモリ454のライトタイミングを説明するためのタイムチャートである。

【図23】図14のデータメモリ454のリードタイミングを説明するためのタイムチャートである。

【図24】CPU30が保有するアドレス空間に対応するメモリマップを示す図である。

30 【図25】CPU30のメモリマップ上において、マスクROM72を割り付け可能な領域を示す図である。

【図26】CPU30のメモリマップ上において、疑似SRAM74を割り付け可能な領域を示す図である。

【図27】CPU30のメモリマップ上において、フラッシュメモリA76を割り付け可能な領域を示す図である。

【図28】CPU30のメモリマップ上において、フラッシュメモリB106を割り付け可能な領域を示す図である。

40 【図29】本実施例において、衛星データ放送受信時の信号の流れを示す図である。

【図30】本実施例の衛星データ放送受信システムで採用されるデータ伝送の階層モデルを示す図である。

【図31】図10のROMデコーダ160において、マスクROM72のチップイネーブル信号CE2Bがアクティブ（ローレベル）になる条件を示す図である。

【図32】図10のPSRAMデコーダ162において、疑似SRAM74のチップイネーブル信号CE1Bがアクティブ（ローレベル）になる条件を示す図である。

【図33】図10のフラッシュデコーダ164において、フラッシュメモリA76のチップイネーブル信号CE3Bがアクティブ（ローレベル）になる条件を示す図である。

【図34】図10のメモリユニットデコーダ166において、フラッシュメモリB106のチップイネーブル信号CE4Bがアクティブ（ローレベル）になる条件を示す図である。

【図35】本実施例において、データ放送を受信する場合に好適なメモリ配置を示す図である。

【図36】本実施例において、フラッシュメモリA76内に記憶されたゲームプログラムを動作させる場合に好適なメモリ配置を示す図である。

【図37】本実施例において、図35のメモリ配置でデ \*

\* ータを受信し、受信後、図36のメモリ配置でデータを表示（または、データをプログラムとして実行）する場合の動作を示すフローチャートである。

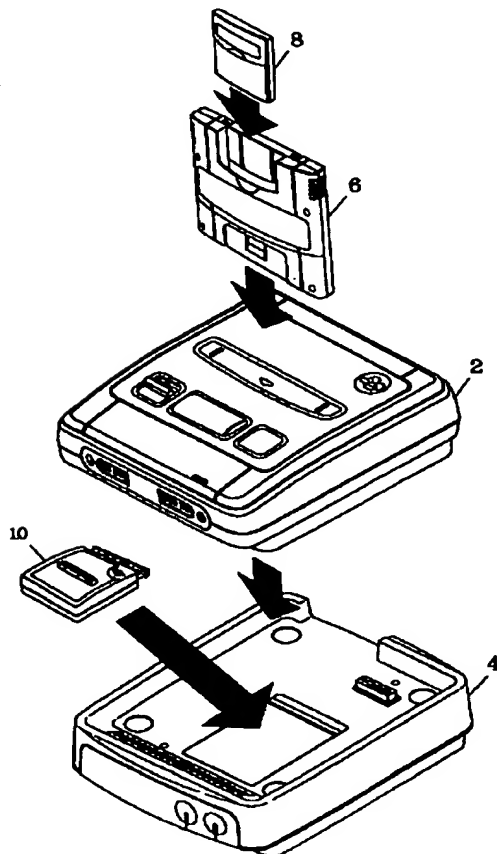
【図38】図37におけるサブルーチンステップS103またはS108において、CPU30によって実行される動作を示したフローチャートである。

【図39】図7に示すマップデコーダ120の他の構成例を示すブロック図である。

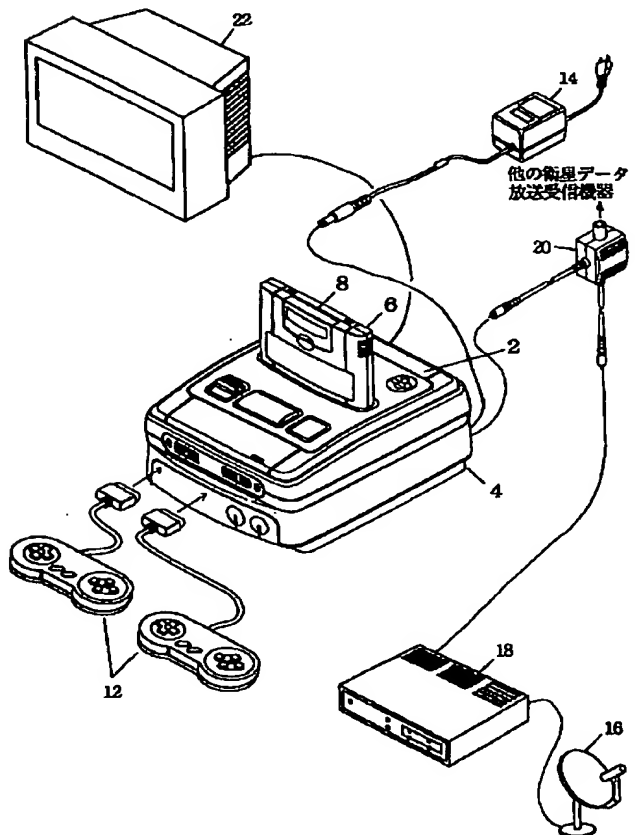
【符号の説明】

- 10 … ゲーム機本体
- 4 … 衛星データ放送受信アダプタ
- 6 … 衛星データ放送受信カートリッジ
- 8 … メモリユニット8
- 10 … ハードディスク装置

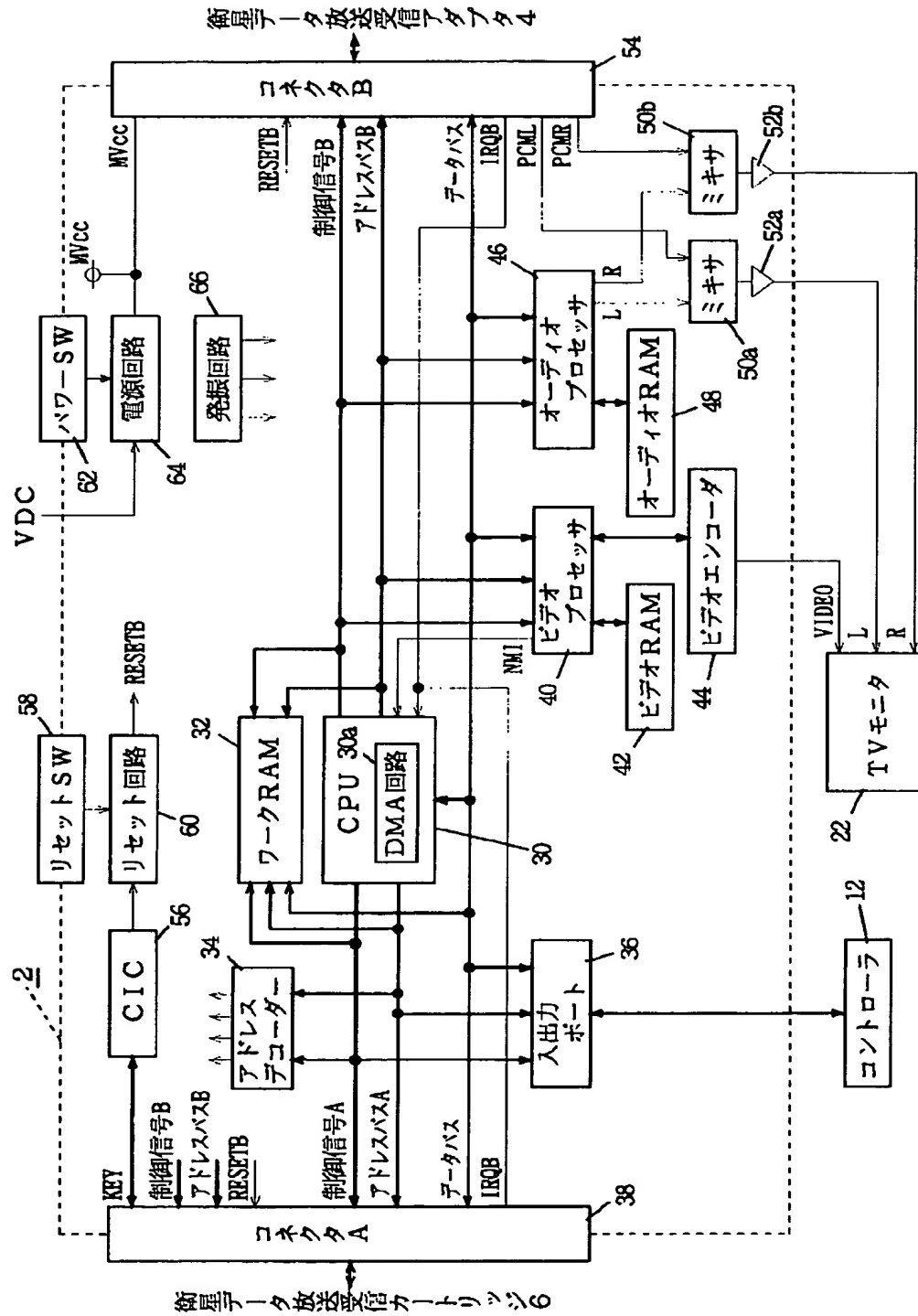
【図1】



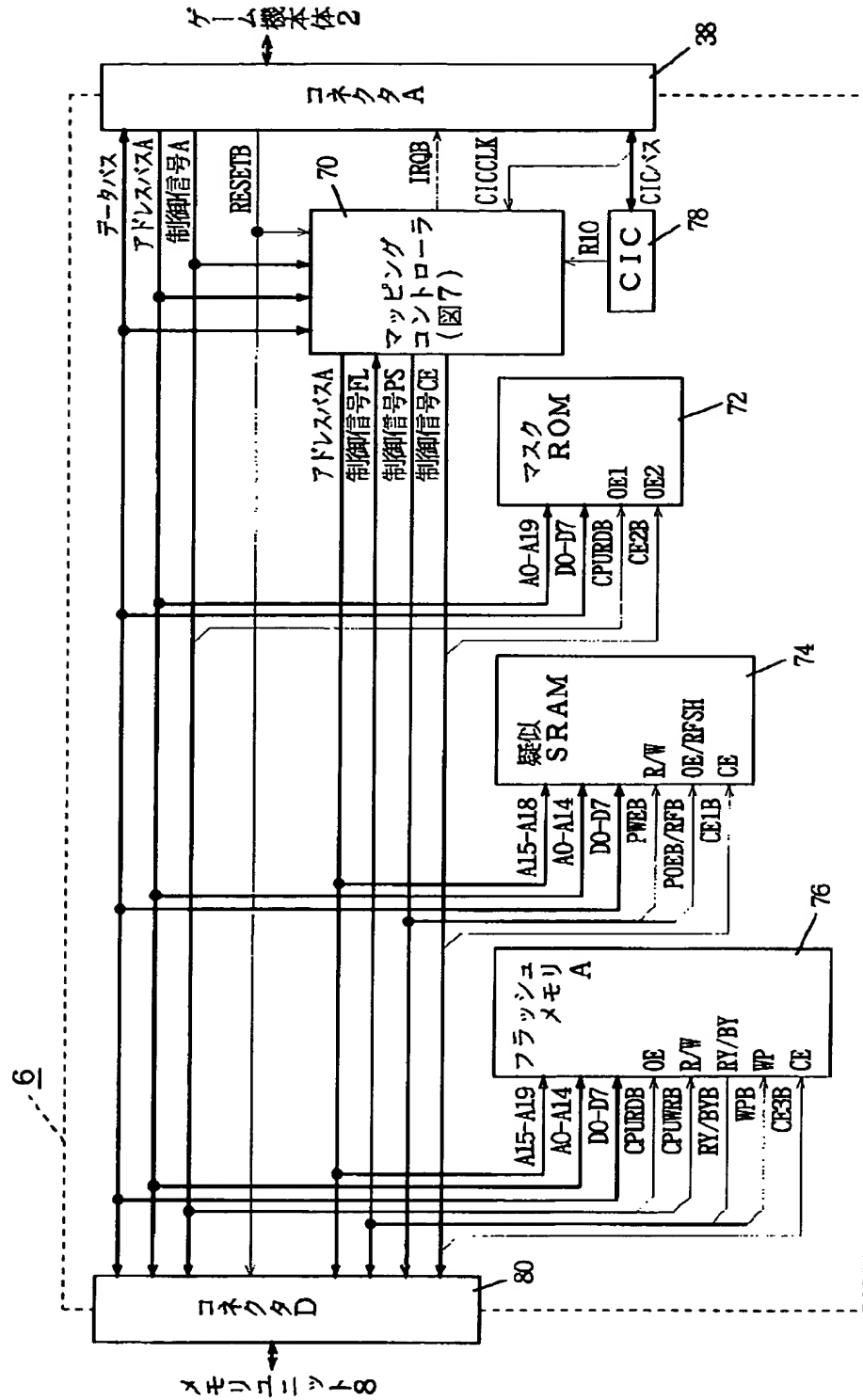
【図2】



【図 3】

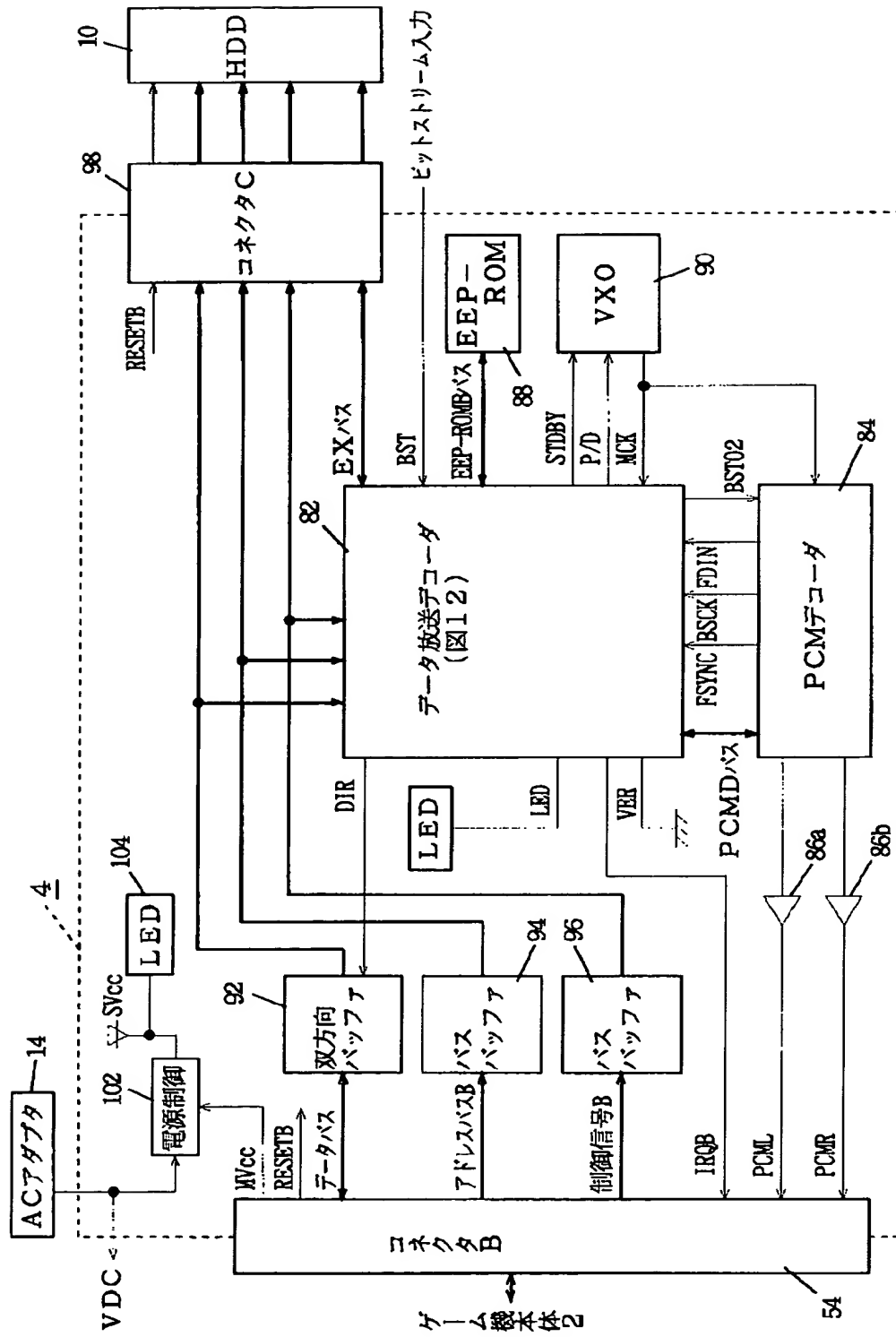


【図4】

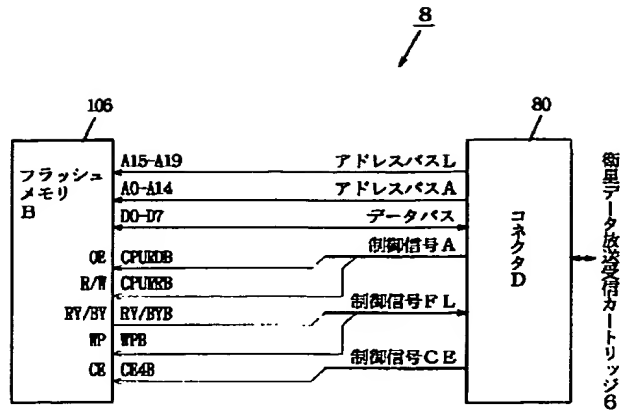




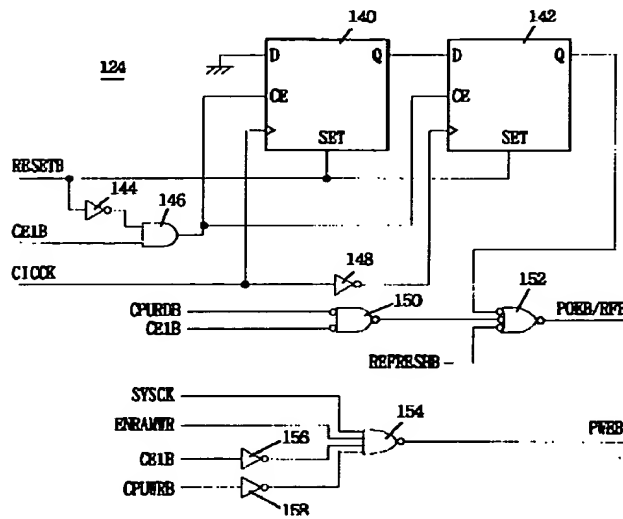
【図5】



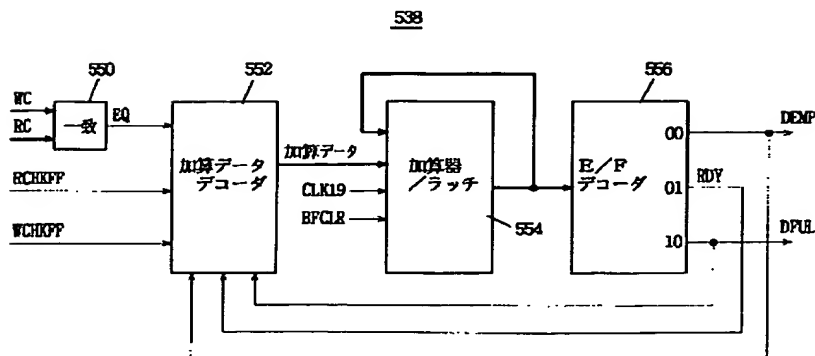
【図6】



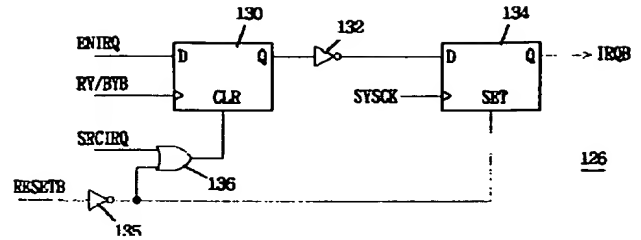
【図9】



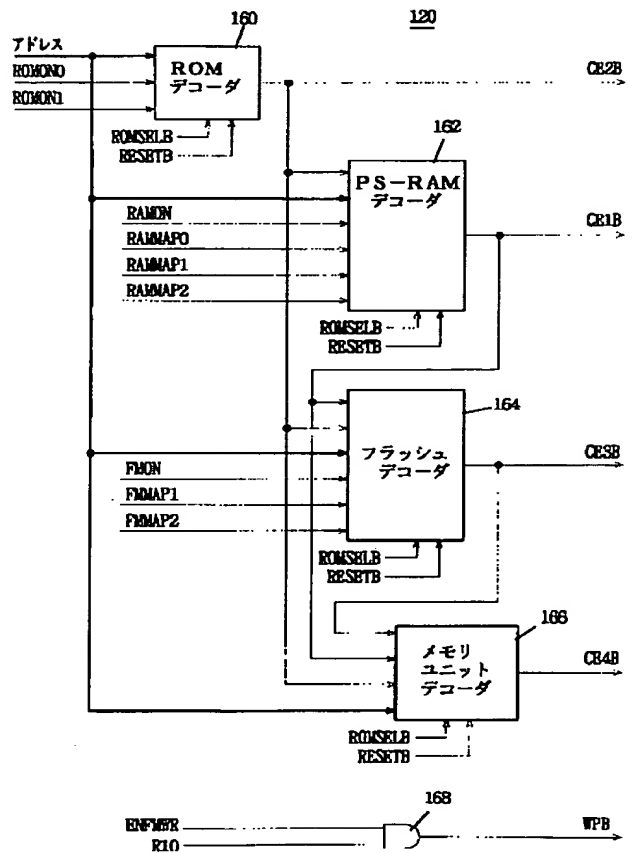
【図19】



【図8】



【図10】



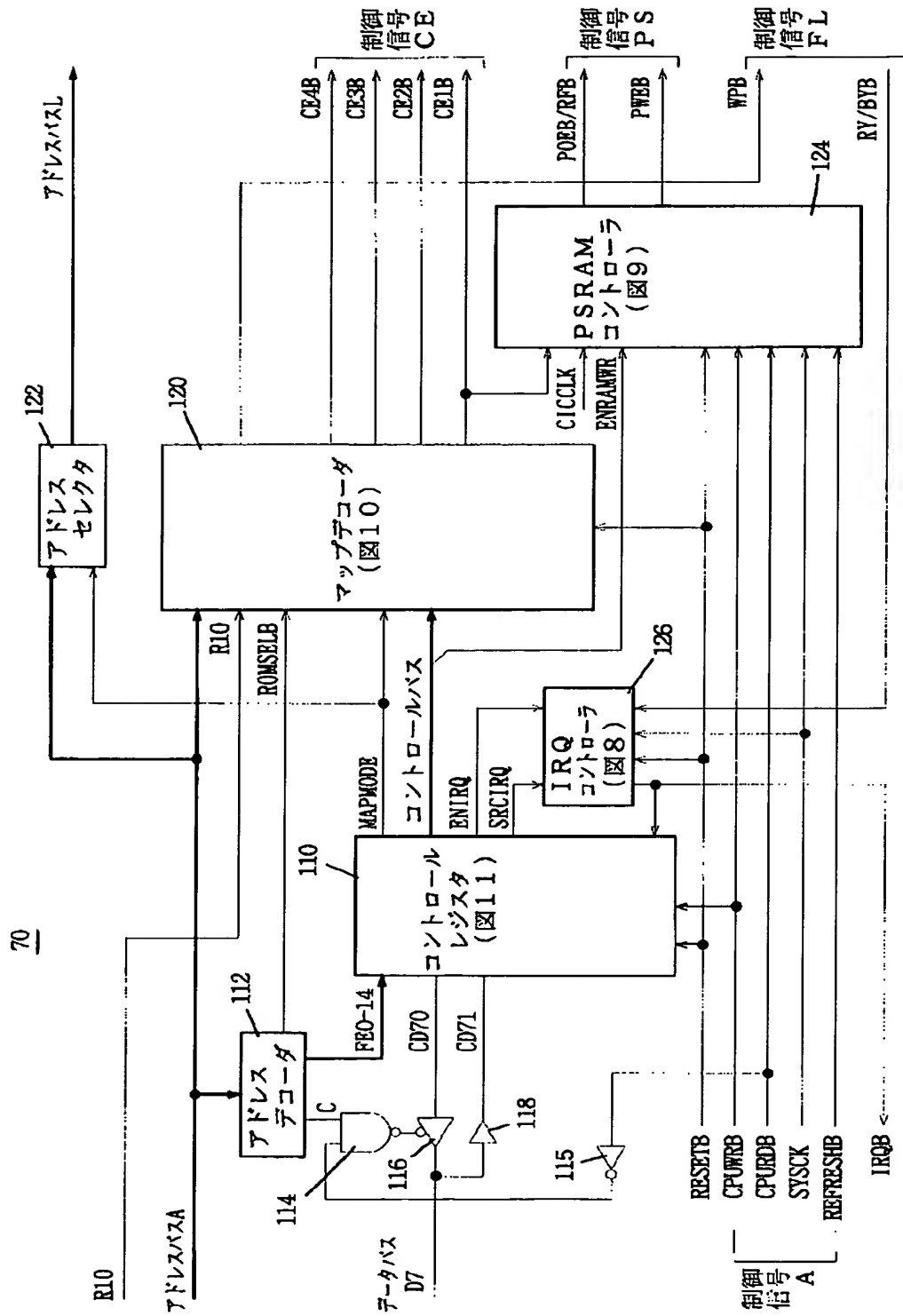
【図31】

RESETB	ROMSELB	A23	A22	A21	ROM0N1	ROM0N0
1	0	0	0	x	x	1
		1	0	x	1	x

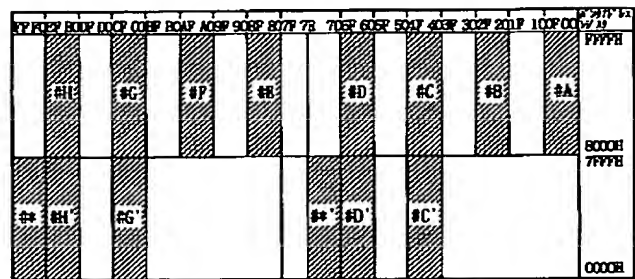
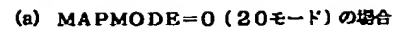
【図34】

RESETB	ROMSELB	POEB	CE2B	CE3B
1	0	1	1	1

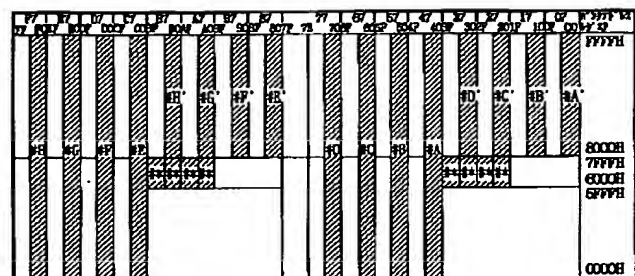
【図7】



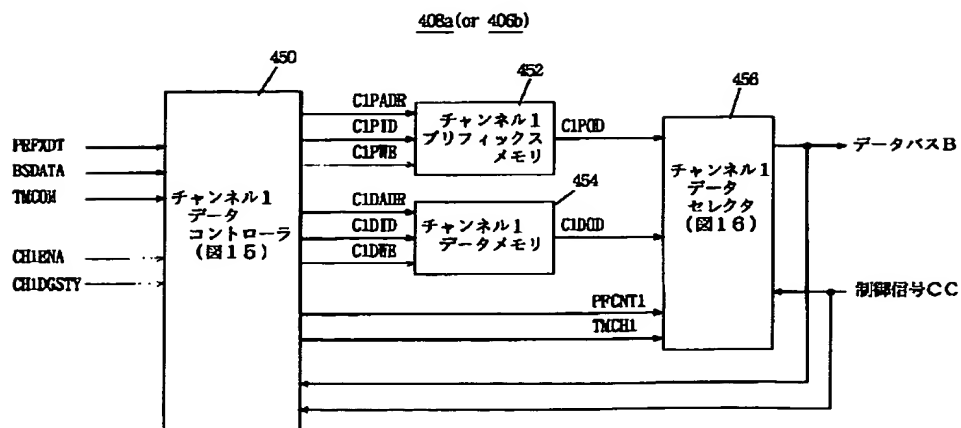
【图 2 6】



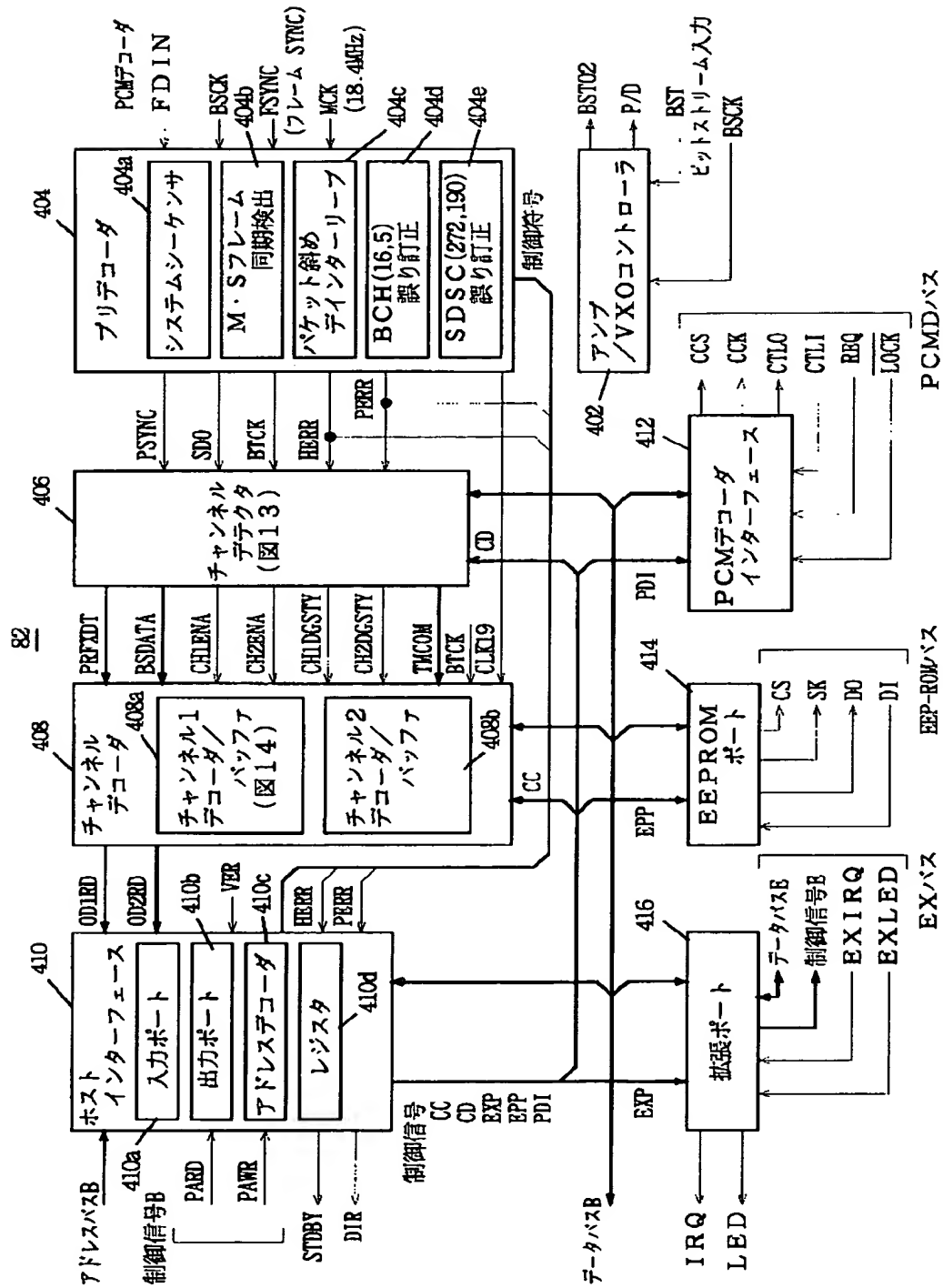
(b) MAPMODE=1 (21モード) の場合



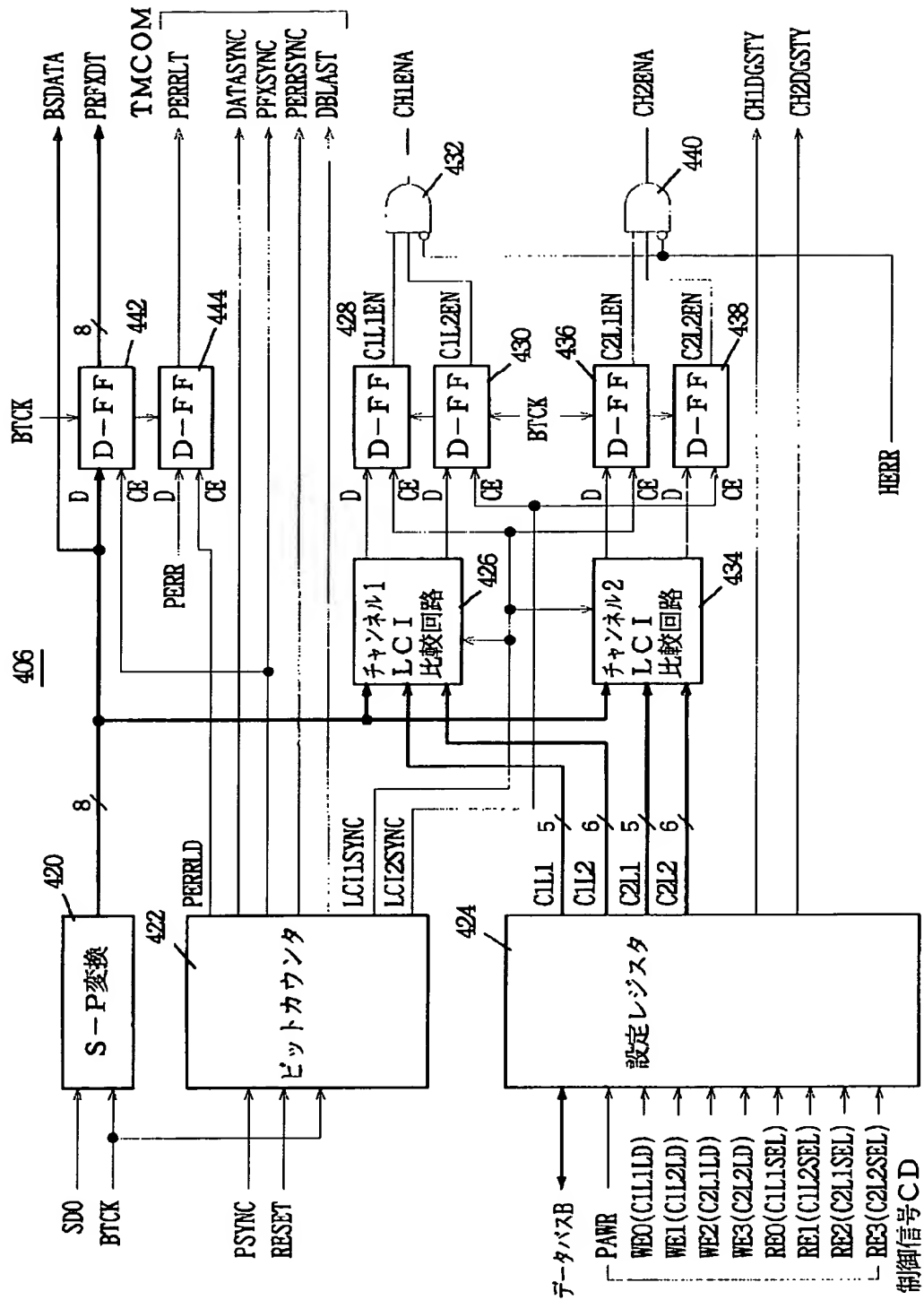
【圖 14】



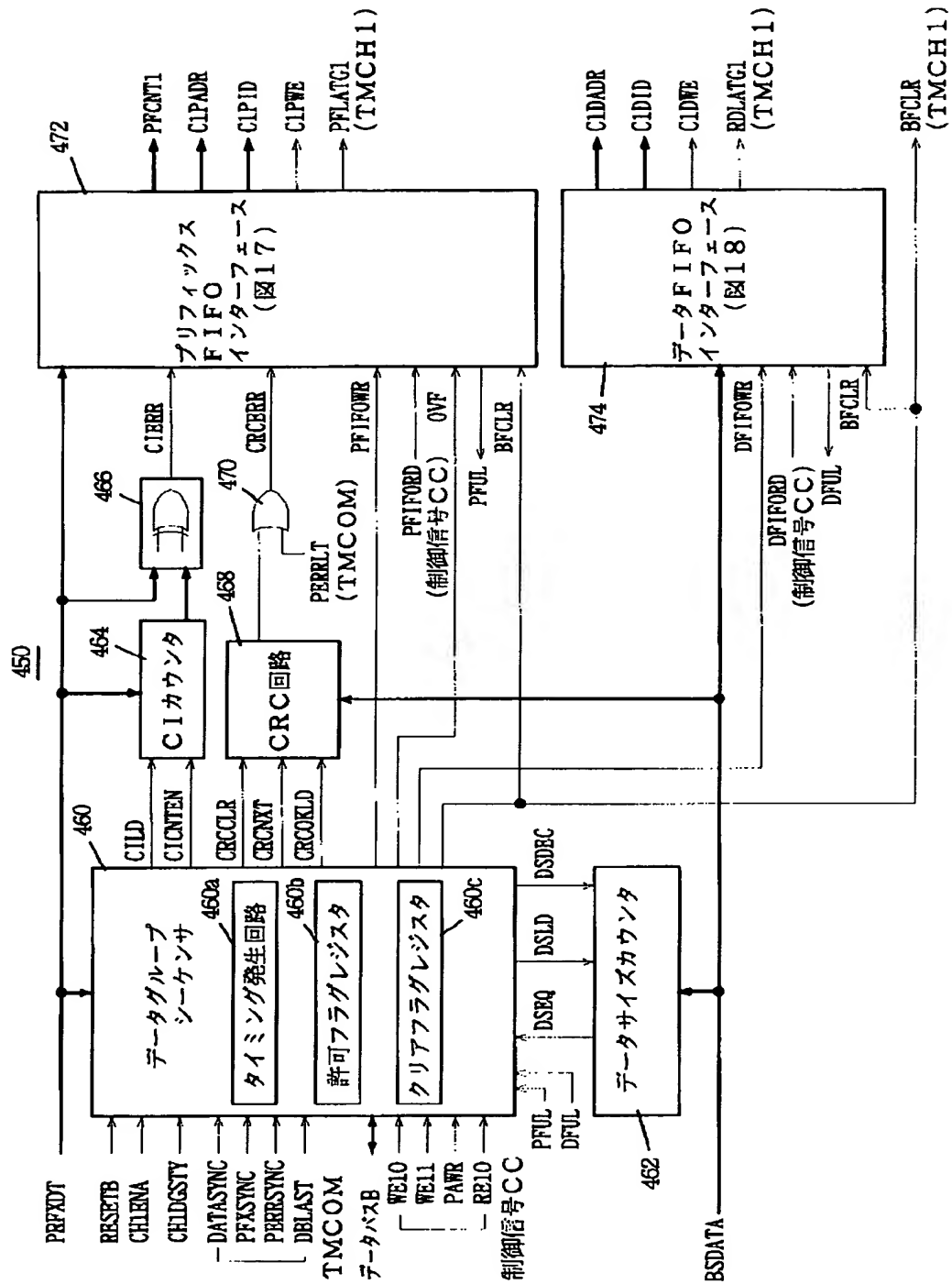
【図12】



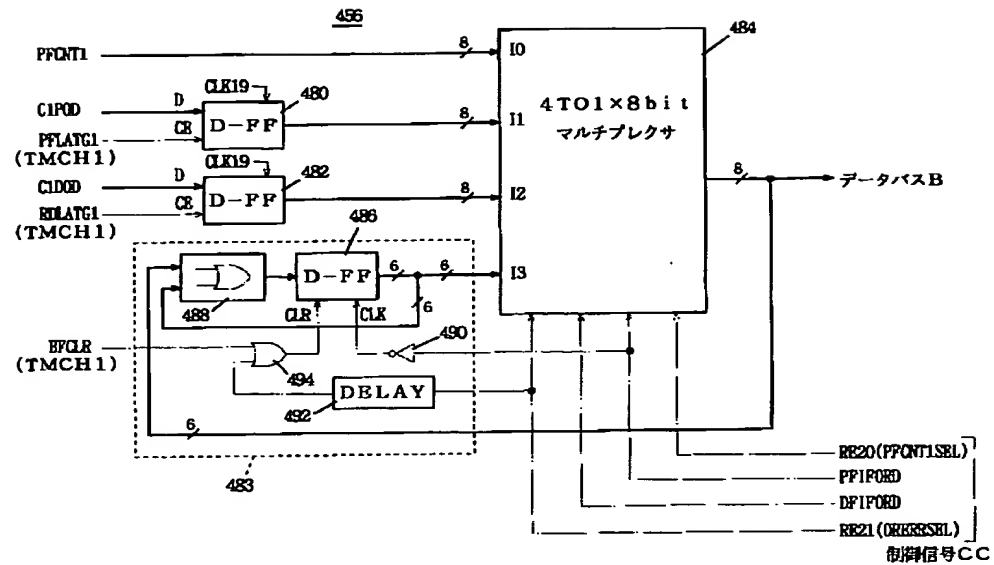
【図13】



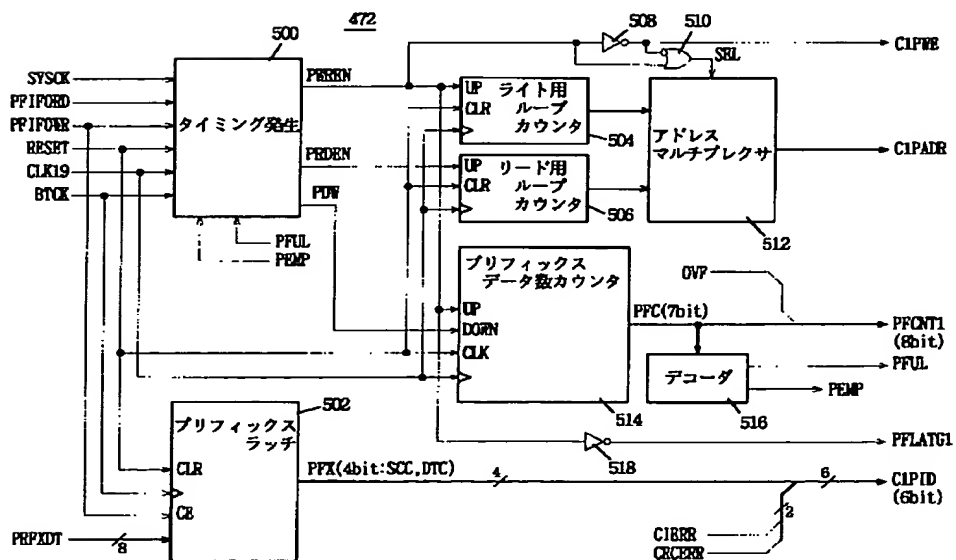
【図15】



【図 16】



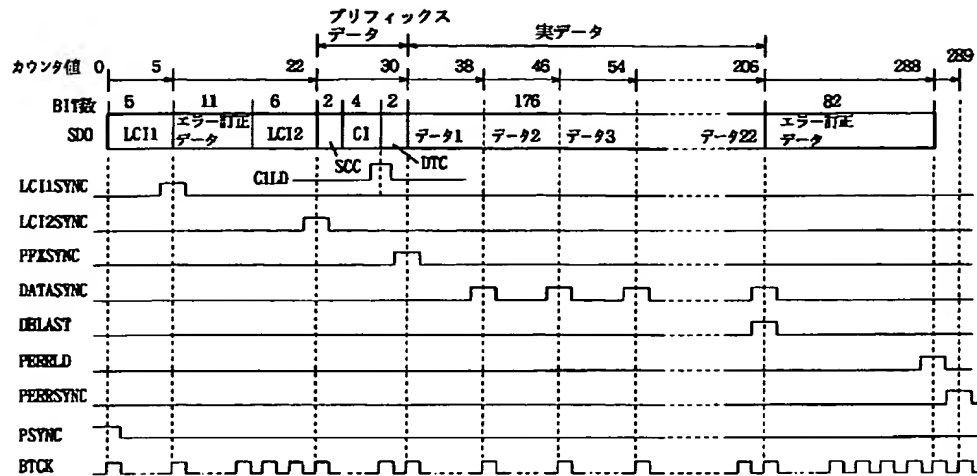
【図 17】





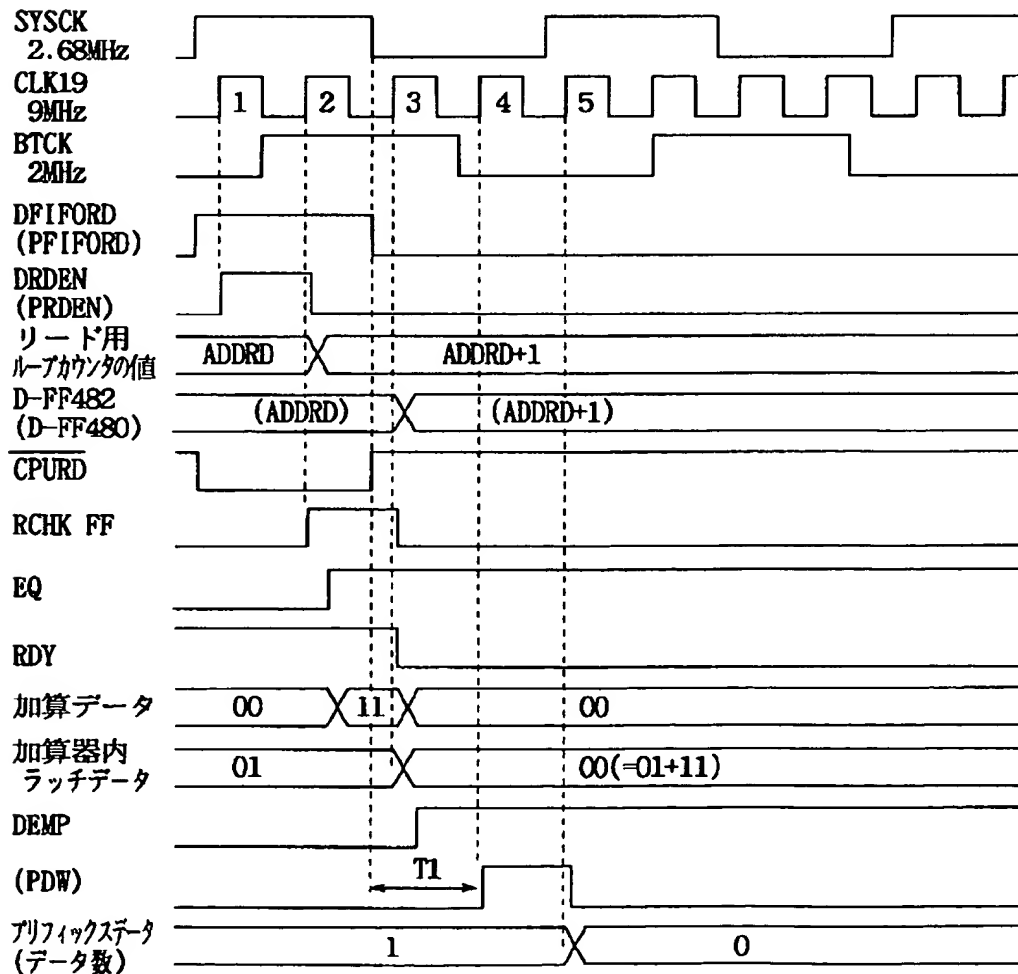


【図 2 1】



【図 2 3】

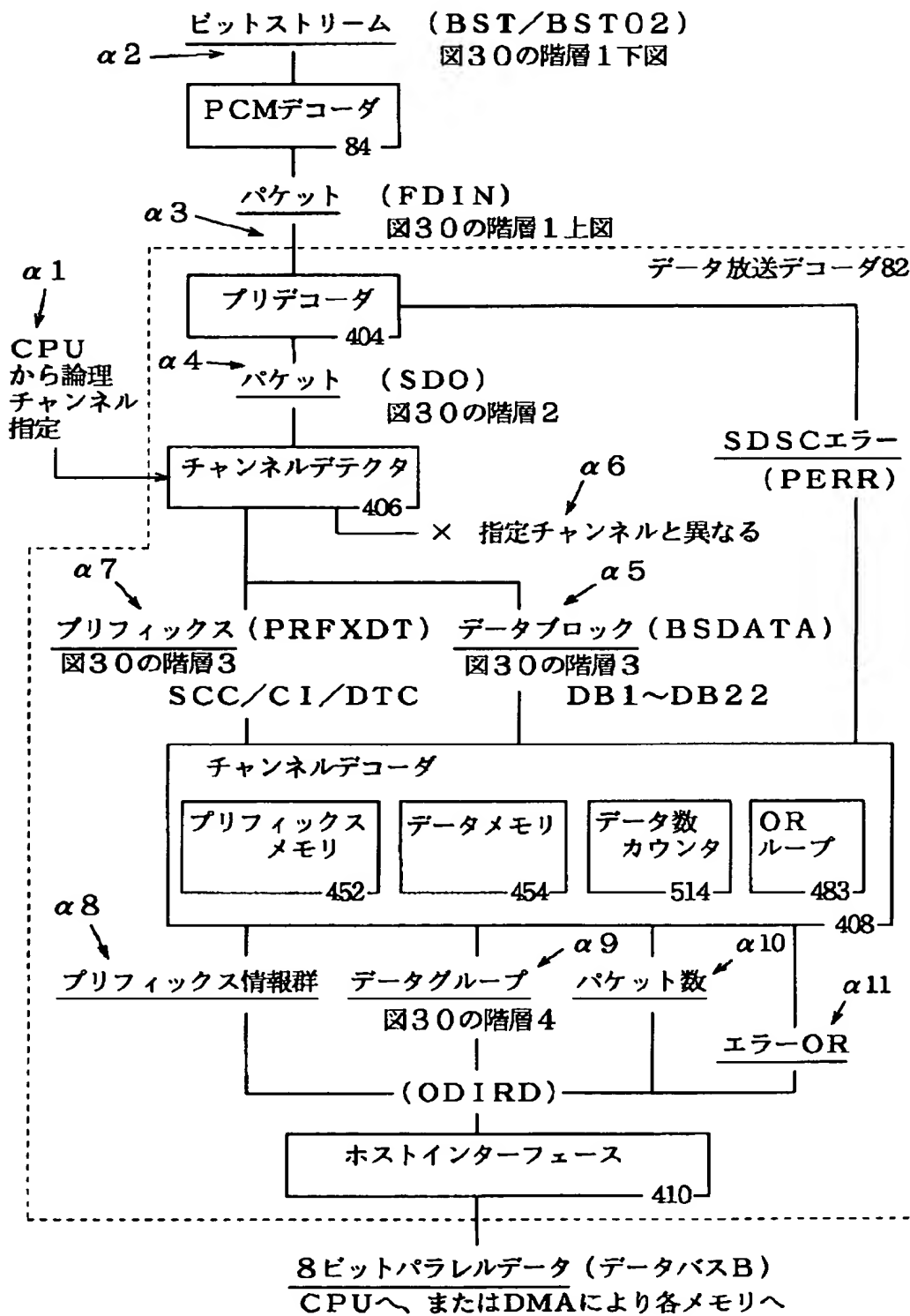
## データメモリアドレスタイミング



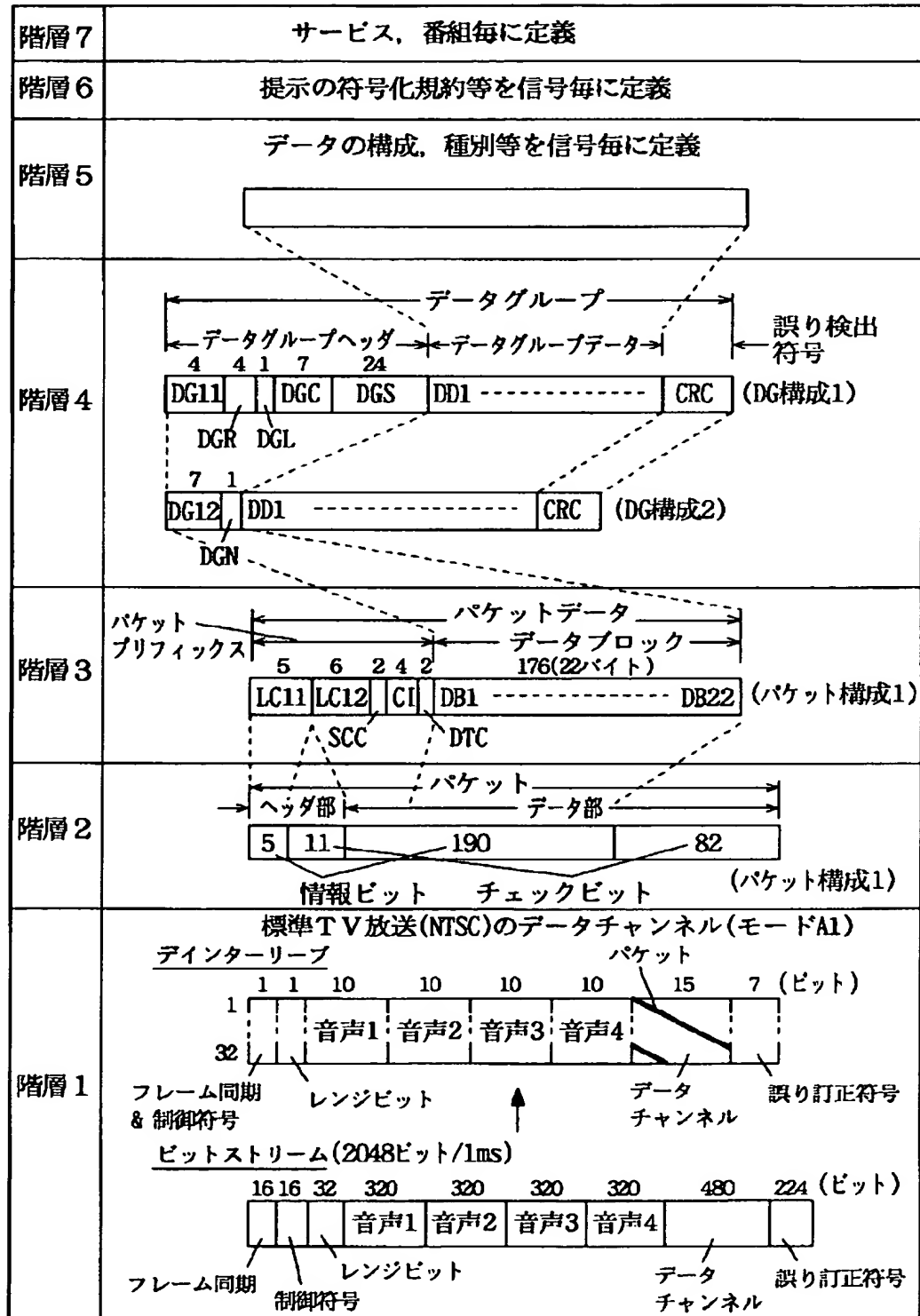




【図29】



【図30】

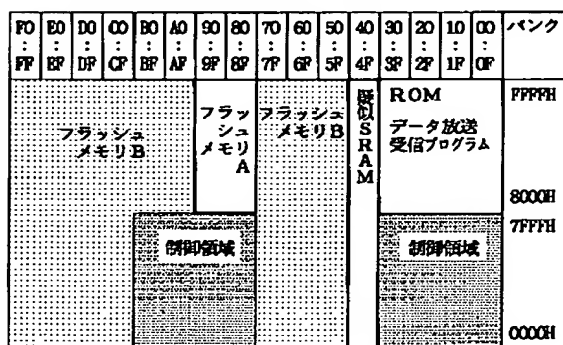


【図32】

RESETB	CE2B	RAMON	MAPMODE	ROMSELB	RAMMAP2	RAMMAP1	RAMMAP0	A23	A22	A21	A20	A19	A15	A14	A13
1	1	1	0	0	0	0	0	0	0	0	0	x	x	x	x
					0	0	1	0	0	1	0	x	x	x	x
					0	1	0	0	1	0	0	x	x	x	x
					0	1	1	0	1	1	0	x	x	x	x
					1	0	0	1	0	0	0	x	x	x	x
					1	0	1	1	0	1	0	x	x	x	x
					1	1	0	1	1	0	0	x	x	x	x
					1	1	1	1	1	1	0	x	x	x	x
			1	0	x	x	x	x	1	1	1	x	0	x	x
					0	0	0	0	x	0	0	0	x	x	x
					0	0	1	0	x	0	1	0	x	x	x
					0	1	0	0	x	1	0	0	x	x	x
					0	1	1	0	x	1	1	0	x	x	x
					1	0	0	1	x	0	0	0	x	x	x
					1	0	1	1	x	0	1	0	x	x	x
					1	1	0	1	x	1	0	0	x	x	x
					1	1	1	1	x	1	1	0	x	x	x
				1	x	x	x	x	0	1	x	x	x	1	1

【図35】

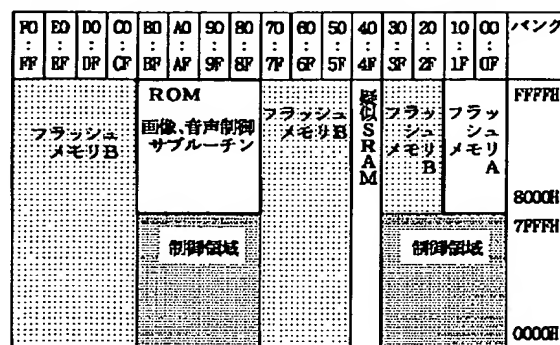
データ放送を受信する場合に好適なメモリ配置



ROMON0=1    RAMON=1    FMON=1  
 ROMON1=0    RAMMAP0=0    FMMAP0=0  
 MAPMODE=0    RAMMAP1=1    FMMAP1=1  
               RAMMAP2=0

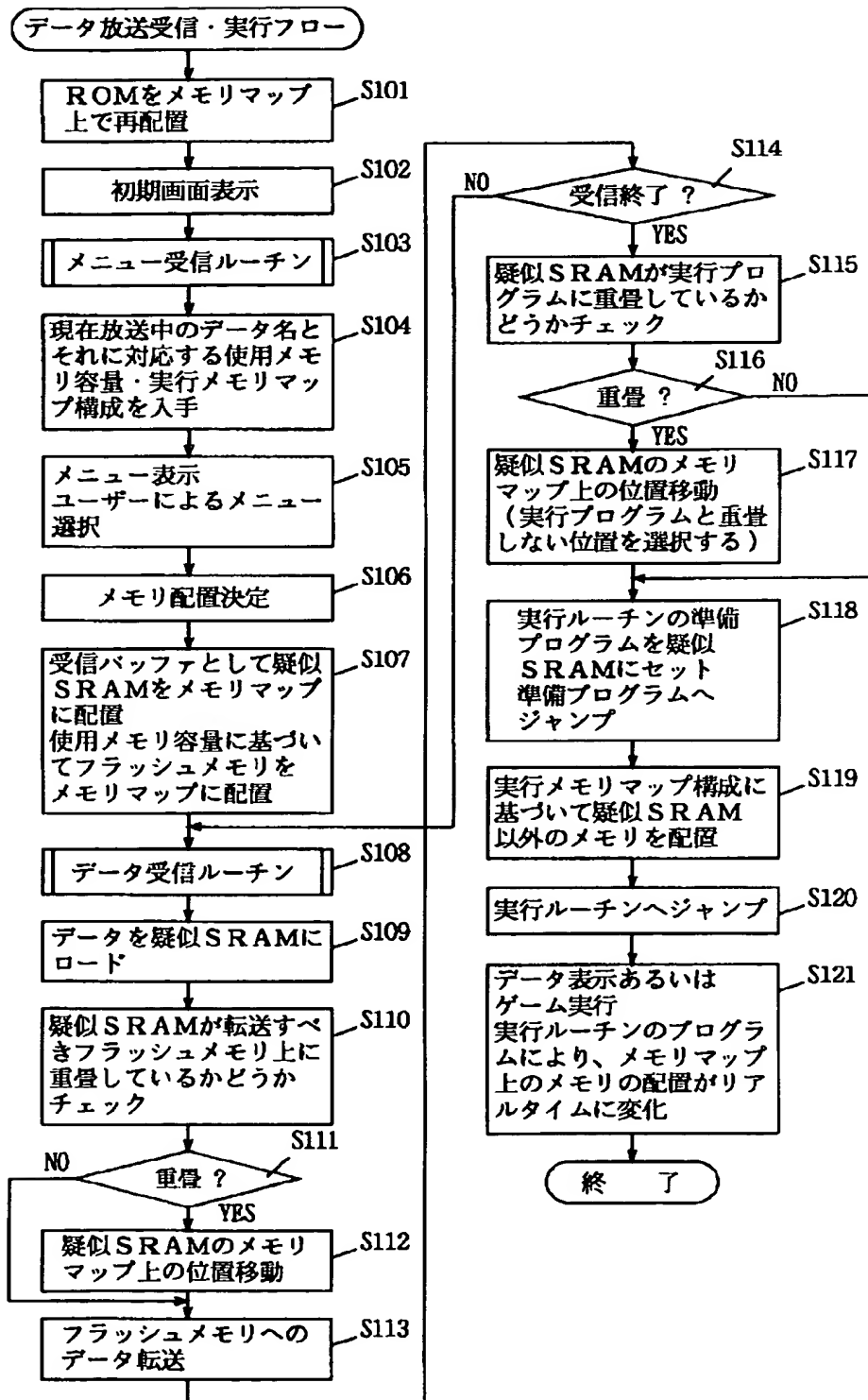
【図36】

受信したゲームプログラムを実行するのに好適なメモリ配置



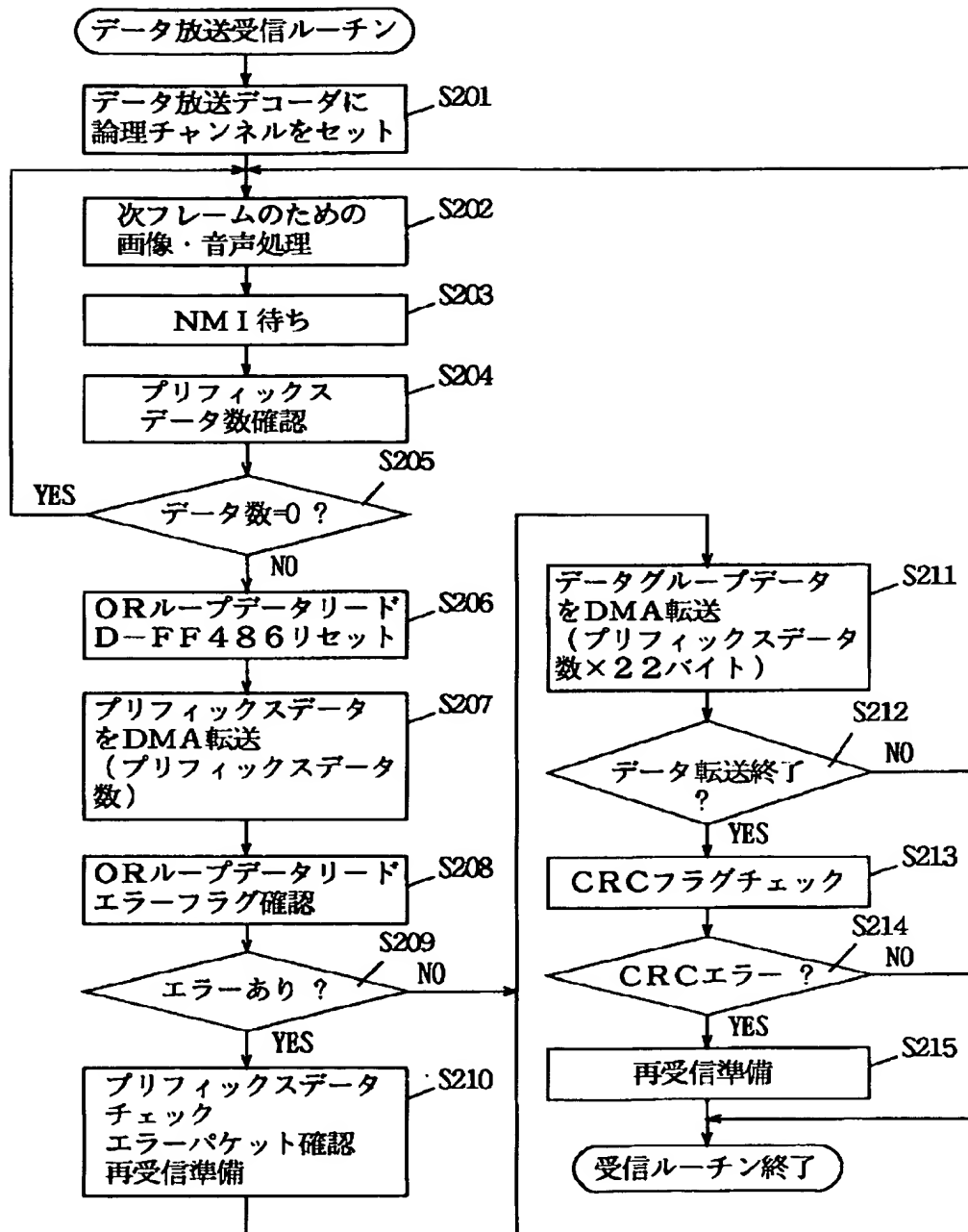
ROMON0=0    RAMON=1    FMON=1  
 ROMON1=1    RAMMAP0=0    FMMAP0=0  
 MAPMODE=0    RAMMAP1=1    FMMAP1=0  
               RAMMAP2=0

【図37】

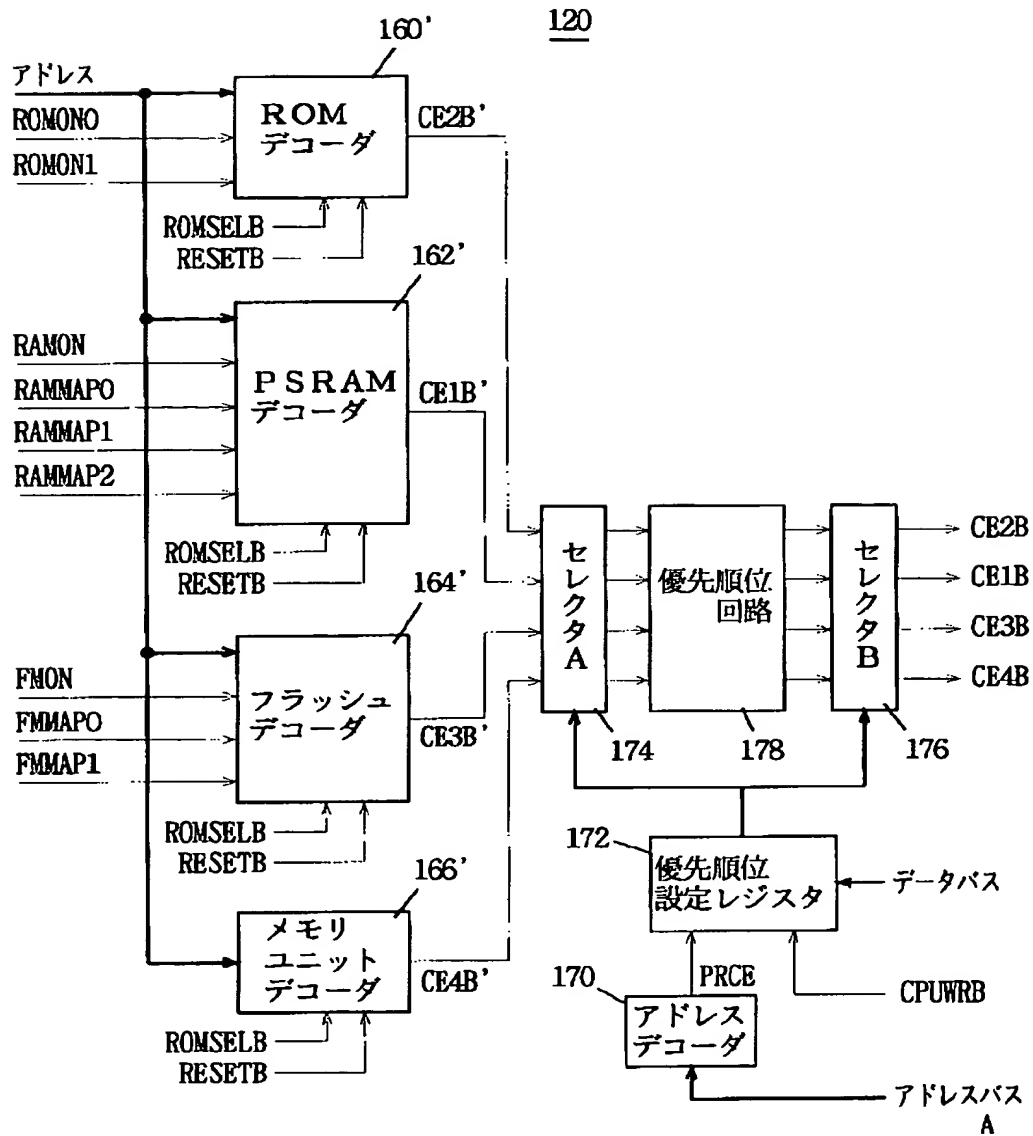




【図38】



【図 39】



フロントページの続き

(51) Int. Cl. 6

H 0 4 B 1/16

H 0 4 H 1/00

H 0 4 L 1/00

識別記号

庁内整理番号

F I

H 0 4 H 1/00

H 0 4 L 1/00

技術表示箇所

H

B

F

(72) 発明者 越智 圭一

兵庫県川西市中央町 3 番 6 号 川西太陽ビル 4 F 株式会社クリエイティブ・デザイン内